Supervisory Control of an Autonomous Underwater Vehicle Using an Acoustic Communication Link

by

William Ryan Kreamer

S.B. in Ocean Engineering
Massachusetts Institute of Technology, 1998

Submitted to the Department of Ocean Engineering
in partial fulfillment of the requirements for the degree of

Master of Science in Ocean Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2000

Author ....
Department of Ocean Engineering
October 1, 1999

Certified by ....
James G. Bellingham
Lecturer, Department of Ocean Engineering
Thesis Supervisor

Accepted by ....
Nicholas Patrikalakis
Chairman, Department Committee on Graduate Students

Supervisory Control of an Autonomous Underwater Vehicle Using an Acoustic Communication Link

by

William Ryan Kreamer

Submitted to the Department of Ocean Engineering
on October 31, 1999, in partial fulfillment of the
requirements for the degree of
Master of Science in Ocean Engineering

Abstract

In this thesis, I designed and tested a supervisory control scheme for the Odyssey II-class Autonomous Underwater Vehicles that relies on a very-low-data-rate acoustic communication link. A human supervisor communicates with the AUV over a combination radio/acoustic network. The supervisor radios commands from shore to data repeater nodes moored at strategic locations on the ocean surface. Utility Acoustic Modems mounted on the moorings rebroadcast the binary data into the sea in the 12-17 kHz frequency band. The moving AUV detects the transmission, decodes the message, and carries out the command contained within. The operator's commands are implemented in the context of a behavior-based layered control software architecture. The supervisory control scheme was tested and verified during the Synaptic Internal Tide Experiment, which took place in Monterey Bay during August and September, 1999.

Thesis Supervisor: James G. Bellingham
Title: Lecturer, Department of Ocean Engineering

Acknowledgments

I gratefully acknowledge the patience and assistance of my advisor, Dr. James G. Bellingham, and the support of Chryssostomos Chryssostomidis, the Director of the MIT Sea Grant Program and the Head of the Department of Ocean Engineering. Thanks are also due to Dr. Mark Johnson, Peter Koski, Aaron Marsh, and Bryan Halay for support with equipment and advice during my work.

This thesis is dedicated to my mother and father, who have been an unending, unquestioning source of support and love. To my grandparents, Lois and Irwin, thank you for your advice, encouragement, and not least, financial support. To Jen, my love, thank you for putting up with me while I was driving myself crazy with school woes. To Yanwu Zhang, who was a daily example of hard work and will power, to Jim Czarnowski, who was my first graduate student role model, and to Sam Tolkoff, who showed me how to finish my thesis, thank you.

Contents

1 Introduction **10**

1.1 Working with the Odyssey II AUV 10

1.1.1 Vehicle Hardware 10

1.1.2 Operational Problems 11

1.2 Supervisory Control 13

1.2.1 System Requirements for Supervisory Control 14

1.2.2 Using an Acoustic Communication Link for Supervisory Control 16

1.2.3 Current Uses of Acoustic Command and Data Links 18

1.3 Current Work 19

1.3.1 Chapter Preview 19

2 The Challenges of Acoustic Communications **21**

2.1 Introduction 21

2.2 Modulation Techniques 21

2.3 Signal-to-Noise Ratio 23

2.4 Propagation Latencies 24

2.5 Multipath Propagation 25

2.6 Communication Link Reliability 26

2.7 WHOI's Algorithms for Robust Transmission and Reception 28

2.7.1 Transmission Algorithm 29

2.7.2 Receiver Algorithm 31

3 Using the Layered Control Architecture for Supervisory Control 33

3.1 Introduction 33

3.2 Behaviors and Arbitration 34

3.3 Modifying a Layered Control Mission 37

4 Field Experiments in Monterey Bay – September, 1999 38

4.1 Introduction 38

4.2 Experiments in Supervisory Control Using Acoustic Communications 38

4.3 Shore Control Station 40

4.4 RF/Acoustic Data Repeater Moorings 41

4.4.1 WHOI's Utility Acoustic Modem (UAM) 42

4.4.2 Accessing the Repeater Moorings 44

4.5 Vehicle Configuration 47

4.6 Support Ship – the R/V Shana Rae 48

5 Field Experiment Results and Data Set Descriptions 49

5.1 Introduction 49

5.2 Communication Performance 50

5.2.1 Message Detection Ratio 54

5.2.2 Byte-Error Ratio 58

5.2.3 Data Transfer Rate 59

5.2.4 Networking Issues 61

5.3 Supervisory Control Performance 62

5.3.1 Command Execution 63

5.3.2 Operator Error 65

5.3.3 Operational Limitations Due to Low Data Transfer Rate and Propagation Latencies 65

5.3.4 Unexpected Layered Control Behavior 67

6 Conclusions and Recommendations for Future Work 69

6.1 Recommendations for Future Work 70

6.1.1 Information Displays 70

6.1.2 Vehicle-to-Vehicle Communication and Networking 70

A Tables **72**

B List of Acronyms **82**

List of Figures

1-1 The Odyssey II AUV. 11

2-1 Block diagram of inner and outer control loops. 25

2-2 Multipath propagation. 27

2-3 Block diagram of transmit and receive signal algorithms for the UAM, reproduced from ref. 17. 28

3-1 Sample mission file, from mission A9925518. 36

4-1 Overview of the RF/acoustic communication network implemented in Monterey Bay. 39

4-2 Locations of data repeater moorings M2, M3, and M4 in Monterey Bay (marked as NOPP 3,1, and 2). 43

4-3 Logical organization of UAM program scripts for the data repeater moorings. 45

4-4 Log file of acoustic messages received at mooring M3. This log file is 25575326.M3. 46

4-5 The AUV was deployed from the R/V Shana Rae, a fine vessel run by Captain Jim Christmann and Angie Christmann of Monterey Canyon Research Vessels, Inc. 48

5-1 Sound speed profile from data taken during mission A9925608. 52

5-2 Histogram of messages sent by the AUV with respect to depth and horizontal range. These figures give an overview of where the AUV spent most of its time underwater. 53

5-3 Two missions that had very low message detection rates. The trajectories are marked with green asterisks to indicate shore-bound messages that were successfully detected by the moorings and red X's to indicate missed messages. 55
5-4 Relationship of 3-D range to MDR. 57
5-5 Two-dimensional histogram relating depth and range to MDR. White space indicates areas where no data exists. 58
5-6 Distribution of byte-errors in each message. Messages with no errors were not included. 59
5-7 Measured heading from mission A9925528. Note the periodic heading disturbance, attributable to the power draw required during UAM transmissions. 62
5-8 The upper plot shows the commanded heading in green and the estimated heading in blue. Note the step-change in commanded heading at 280 seconds. 63
5-9 The first mission successfully demonstrating operator-control of the AUV (figures not to scale). 64
5-10 Mission showing the effects of operator carelessness. The operator had planned for the AUV to turn left to 260°, but used the wrong units in the command. Instead the AUV turned right to 137°. 66
5-11 This mission was a long run out the axis of Monterey Canyon. Note the unexpected end of mission indicated by the absence of a return-to-surface trajectory. 68

List of Tables

2.1 A sample QFSK modulation alphabet. 23

2.2 UAM Header Structure. 29

4.1 Location of data repeater moorings in Monterey Bay. 42

5.1 Vehicle mission information, with associated mooring log files. 51

A.1 Communication Statistics for mission A9925518. 73

A.2 Communication Statistics for mission A9925520. 74

A.3 Communication Statistics for mission A9925523. 75

A.4 Communication Statistics for mission A9925526. 76

A.5 Communication Statistics for mission A9925528. 77

A.6 Communication Statistics for mission A9925608. 78

A.7 Communication Statistics for mission A9925811. 79

A.8 Communication Statistics for mission A9925812. 80

A.9 Communication Statistics for mission A9925813. 81

Chapter 1

Introduction

1.1 Working with the Odyssey II AUV

The Odyssey II underwater vehicle is a second-generation survey-class autonomous robot designed for use as an intelligent mobile instrument platform [4]. It is an extremely robust and successful design, with over 400 dives in field deployments since 1992 in Antarctica, the Arctic, the Juan de Fuca ocean ridge in the Pacific Northwest, the Haro Straits off of British Columbia, Monterey Bay, New Zealand, the Labrador Sea, Massachusetts Bay, the Mediterranean, and the Gulf of Mexico.

1.1.1 Vehicle Hardware

Housed in a low-drag fairing with a single propeller and cruciform control surfaces, the Odyssey II AUV is 2.2 meters long (see Figure 1-1). It has a maximum diameter of 0.6 meters, and a maximum speed of about 3.5 knots. The fairing is free-flooded and contains the main pressure housings, which are two glass spheres. In the present configuration, the vehicle has an endurance of eight to twelve hours, depending on the operating speed and the usage of the various power-consuming subsystems, such as the acoustic modem.

The primary onboard computer is built around a Motorola 68030 microprocessor. In addition to the main computer, a network of small micro-controllers is used to

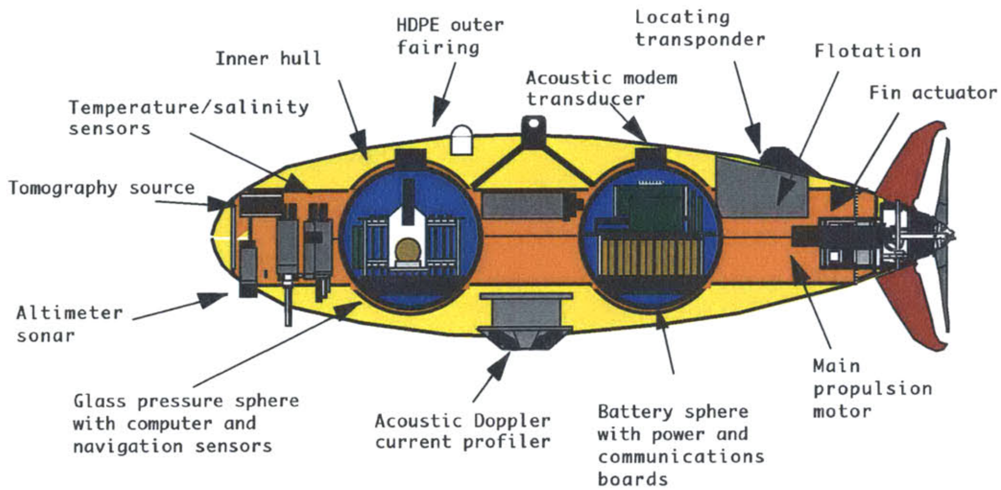


Figure 1-1: The Odyssey II AUV.

distribute "intelligence" to sensors and actuators. In its present state, the vehicle sensor complement includes a three-axis gyrocompass, three-axis accelerometer and angular rate sensors, an altimeter, and a pressure transducer. Scientific sensors and instruments include conductivity and temperature sensors, an acoustic doppler current profiler, a Utility Acoustic Modem (UAM) from WHOI, and a hydrophone and source for acoustic communications[1].

Vehicle operators may use an acoustic transponder, a radio beacon, or a strobe to locate the vehicle. All three locating aids operate from power sources independent from each other and from the rest of the vehicle electronics, to ensure operation even if the vehicle batteries are low. Operators use the radio beacon and strobe for locating the vehicle on the surface, and can track the vehicle under water using an ultrashort-(USBL) Trackpoint II system from a support ship.

1.1.2 Operational Problems

This thesis is motivated partly by operational failures that the Sea Grant team experienced during the Odyssey-class vehicles' many field deployments. While these

[1]Please note that the UAM source and receiver are now located in different positions on the vehicle.

vehicles have had, in general, superlative operational records, Murphy's Law strikes every project sometime, and MIT Sea Grant is no different. Interviews with experts in the unmanned vehicle (UV) field and reviews of MIT Sea Grant Post-Cruise Reports highlight potentially serious operational failures associated with errors in mission programming, inadequate hardware checks, subtle low-level programming errors, and ordinary bad luck. The following accounts of UV failures were taken from MIT Sea Grant Post-Cruise Reports [14, 15], personal correspondence [1, 2, 13, 19, 25, 28, 29], and conference papers [24]:

- An error during mission programming led to the grounding of the MIT Sea Grant vehicle Xanthos off the island of Elba, Italy.

- The tail cone section of one of the Odyssey vehicles was damaged and replaced with the tail cone from another vehicle. Diagnostic tests seemed to indicate satisfactory elevator/rudder operation but failed to reveal a reversal in motor wiring which reversed rudder and elevator commands. Upon release and submersion, the vehicle foundered at the surface for the duration of its one-hour long mission.

- A combination of a software fault and a jammed weight release mechanism caused the Autonomous Benthic Explorer (ABE) to get stuck on the sea floor at a depth of 4100 meters. Recovery was accomplished without further mishap by Deep Sea Vehicle Alvin [30].

- A combination of a subtle low-level programming error combined with insufficient pre-launch checks resulted in the near-loss of the AUV Amphitrite on the bottom of Monterey Canyon in 700 meters of water. Recovery was accomplished by the Remotely-Operated Vehicle (ROV) Ventana.

- A waypoint entered incorrectly during the mission-planning phase sent a Remote Environmental Monitoring UnitS (REMUS) swimming optimistically for New Jersey, 300 miles away. A quick mission abort and a second inspection of the mission plan revealed the error.

- An incorrect value for the local magnetic variation led to a minor 29 degree offset in REMUS heading. (Note: Must have been an incorrect sign. That's about twice the value of the local mag. var.)

The vehicles that survived the problems listed above (ABE, REMUS, Odyssey) all are robust systems, built by experienced, skilled engineers and scientists. Yet, they were each vulnerable to the same system deficiency - once launched, the vehicles were isolated from all but the most rudimentary contact and control. Interaction was limited to acoustically triggering a weight release to force the vehicle to the surface, or other single-use systems. The simple fact is, even though each of these vehicles has made progress toward the goal of reliable, autonomous operation, no machine can be as smart or adaptable as a human. To achieve the most out of "autonomous" systems, we must provide the ability for humans to intervene when the necessity arises.

The ability to observe and intervene in the course of an otherwise automatically-controlled process is often called *supervisory control*. The goal of this work is to use an acoustic communication link to implement a supervisory control capability for the Odyssey II AUV. With such a capability in place, the character of operations would shift from the current "fire and forget" regime to a much more interactive relationship between human operator and AUV.

1.2 Supervisory Control

A long-time expert in the field, T. B. Sheridan describes supervisory control in the following way [23]:

> "In the strictest sense, supervisory control means that one or more human operators are intermittently programming and continually receiving information from a computer that itself closes an autonomous control loop through artificial effectors and sensors to the controlled process or task environment."

He describes a spectrum of control, with manual control (in which the human operator makes continual adjustments in an attempt to control a process) at one end, and fully automatic control (in which the human is merely an observer, and a computer controls the process through actuators and sensors) at the other. In the middle lies a continuum of supervisory control in which the human operator and a computer share varying levels of responsibility in the control of a process or accomplishment of a task.

The original designers of the Odyssey II AUV strove to place the vehicle as far toward the automatic end of that spectrum as possible. In some sense, they succeeded - humans only interacted with the vehicle at the beginning and end of its mission. During the sometimes lengthy missions (as long as six hours), the operators could track the AUV with an acoustic pinger, but, if an emergency arose, there existed no robust capability for human intervention. Vehicle operators need the ability to intervene when they see indications that the automatic controller cannot cope with a developing situation. Consider the following scenario, in which the AUV is below, conducting hydrographic surveys, and the weather on the surface is quickly deteriorating. Not knowing how long the storm might last, perhaps the only way for the operators to guarantee the survival of the vehicle would be to command it to the surface immediately. Without this ability, the ship's crew is at the mercy of the storm, forced to attempt a dangerous recovery in unforgiving conditions when the vehicle surfaces at the end of its mission.

1.2.1 System Requirements for Supervisory Control

Classical control theory predicts that time delays in a feedback control loop can incite potentially disastrous instabilities (try staying in the same lane by looking at the painted road lines only as they pass in your rear-view mirror!). In the sixties, as part of research into the problem of how operators on earth could manipulate robots on the lunar surface through the three-second round trip time delay imposed by the speed of light, researchers showed that remote manual manipulation was safe only by operating in a "move-and-wait" fashion [9]. The operator would command as large a movement as possible without risking collision or other error and then wait three seconds (one round-trip time delay) for feedback about the result of his command. The delay forced operators to perform maneuvers in small increments, making even simple tasks unacceptably (but unavoidably) time-consuming and tedious.

Solving the Delay Problem

In more modern systems, advances in actuators, sensors, software, and computer processors have made it possible to remove humans from the step-by-step details of control. Now, operators can issue more abstract, less-frequent, goal-oriented commands that a semi-intelligent subordinate system then carries out. In the case in which the computer that closes the subordinate control loop is collocated with the remote device, there is no delay, and thus no instability. The operator, though remote, can review the progress of the task, revise commands, and step in as needed.

Of course, this scenario cannot eliminate the signal propagation delay between the operator and the vehicle computer, nor does it address communication bandwidth requirements. Specific calculations of the maximum allowable delay and required bandwidth are directly affected by the dynamics of each system, but we may use the following criteria as basic guidelines [23]:

1. The operator's command should encompass a relatively large "bite" of the task at hand. Thus, the operator's intervention in the control loop can be at a much-reduced frequency compared with that of the subordinate automatic controller.

2. The unpredictable aspects of the remote environment must not change too rapidly (the environment should have a sufficiently low disturbance bandwidth). If the operator can trust that his next image of the remote environment will look much like his last, then he can plan future actions farther in advance. This is the principle behind the "move-and-wait" method of remote manipulation.

3. The subordinate automatic system should be trustworthy. If the subordinate automatic controller's response to disturbances is robust, more responsibility for the minute-to-minute survival of the system can be transferred to it by the

human operator. Assuming a sufficiently controllable vehicle, a well-designed proportional-derivative controller can mitigate the dangers of an unpredictable environment.

1.2.2 Using an Acoustic Communication Link for Supervisory Control

Acousticians have studied underwater digital communication for many years, and low data rate acoustic telemetry (up to 100 bits per second in benign environments) has been available since the early1990s. With the emergence of high-speed, low cost DSP chip sets in the early 1990s, Datasonics Inc. (of Massachusetts) and researchers at the Woods Hole Oceanographic Institution (WHOI) produced a commercial acoustic modem that could deliver 1,200 bits per second in a half-duplex communication link. The system could be used in either a deep-water, vertical channel or a shallow-water, long-range environment with severe multipath interference [20].

To those readers used to hearing about 56 kilobaud modems and 10 mega-bit-per-second internet connections, these data rates may seem to be incredibly slow. However, this is the reality of underwater acoustic communication; because of the severe attenuation of radio-frequency waves in water, bandwidth limitations are unavoidable for applications requiring even a modest transmission range.

State of the Art

To reduce data errors, use bandwidth and energy more efficiently, and to increase the data transfer rate, researchers have focused on two modulation techniques - one involving phase coherent demodulation, and the other non-coherent demodulation. The two schemes have widely different performance characteristics, and the choice depends on the requirements of the environment and the application.

Phase coherent demodulation techniques offer high data rates (higher than 10,000 bits per second without coding) and excellent bandwidth efficiency, and greatly reduce the transmission time of the data stream [11]. Coherent demodulation is usually implemented through the use of Quadrature Phase Shift Key-ing (QPSK), which requires the receiver to detect both the magnitude and phase of the received signal. Its principal disadvantages are increased processing complexity and cost, and it is most suited for situations that can offer a high signal-to-noise ratio (SNR) – i.e. for short ranges, high source levels, and low ambient noise levels.

Systems that are implemented using non-coherent demodulation techniques offer more reliable transmission, are very resistant to noisy environments and multipath interference, and are easier to implement and deploy, resulting in lower development cost; however, they are limited to much lower data rates. Currently, the maximum raw data rate for commercially-available systems is 2,400 bits per second [7]. Error coding, transmission redundancy, packet delays, and power limitations lower the effective data rate (actual data through-put) even more. Depending on the acoustic environment, the effective data rate can range from 10 bits per second to 2,400 bits per second. Non-coherent demodulation techniques are typically implemented using a Multiple Frequency Shift Key-ing (MFSK) approach, and are generally used when ambient noise levels are high, longer communication ranges are required, or communication link reliability is very important.

Finally, work is ongoing to adapt and develop other modulation techniques for underwater digital communication - Sequence Position Modulation (SPM) is an extension of Pulse Position Modulation (PPM) that permits co-channel, asynchronous, multiple access to a single receiver. Sanchez et al. [21], describe work in which an SPM modem is shown to provide acceptable link quality at 160 bits per second at a relatively noisy 0 dB signal-to-noise-ratio (SNR) in a non-minimum-phase multi-path channel of 5 kHz bandwidth and with a delay spread of 84 milliseconds. This work could be significant, especially for enabling multiple-vehicle operations in littoral and surf-zone waters, a research area of significant contemporary interest and practical significance (underwater mine counter-measures).

1.2.3 Current Uses of Acoustic Command and Data Links

While the expected transmission delays for underwater acoustic communication are comparable to or slightly greater than those in earth-to-moon situations, the data transfer rates are much slower. In spite of this obstacle, engineers and scientists have made significant progress in using underwater acoustic communication links for data telemetry and supervisory control.

Subsea Teleoperation with JASON

Subsea Teleoperation with JASON One example of the effective use of acoustic communication for supervisory control can be found in work done by Sayers et al. [22]. They investigated the feasibility of using an acoustic communication link to perform telemanipulation tasks (grasping) with the JASON ROV. In 1994, the researchers performed experiments with a simulated acoustic link in which an operator in Pennsylvania performed manual grasping tasks with the ROV deployed off of Massachusetts. For simplicity, they assumed that the acoustic link had a bandwidth of 10,000 bits per second, perfect reliability (i.e. no breaks in communication, no errors) and a selectable delay of 5, 10, and 15 seconds. Good results were obtained, suggesting that tasks traditionally performed via manual control (i.e. with high bandwidth, low delay communications) can be accomplished with the appropriate tradeoffs between message size, frequency, and level of abstraction.

Acoustically-Controlled UUV – Hugin

Hugin The Hugin Unmanned Underwater Vehicle (UUV) is a product of the Norwegian Defense Research Establishment (FFI), which developed the vehicle for high-precision deep-water seabed mapping [27]. While the Hugin vehicle has some degree of autonomy, FFI designed it to be continuously supervised and controlled by an operator on a support ship through an acoustic link [26].

The Hugin vehicle uses three acoustic links; the first is a low data-rate command link (55 bits per second, without encoding) with FSK modulation, the second is a dedicated one-way data link (1,400 bits per second) for real-time data quality control, and the third is an emergency bi-directional acoustic link that operates through the acoustic positioning system [18].

1.3 Current Work

The focus of this work is to use an underwater acoustic link to implement a supervisory control capability for the Odyssey II AUV. This goal was met, with advances in the following areas:

- Designed a high-level command language to provide a wide range of capabilities to the operator. Special attention was paid to efficient use of available bandwidth, error-handling, and reliable control.

- Implemented the command language on the Odyssey IIb AUV in the context of the behavior-based layered control architecture. Enabled the operator to make real-time requests for vehicle information during the mission itself.

- Performed field experiments of the supervisory control capability in Monterey Bay, CA. These experiments provided valuable practical experience in using a very low data rate communication link for supervisory control, and highlighted several areas for improvement.

1.3.1 Chapter Preview

The Challenges of Acoustic Communications

The many challenges of communicating through underwater acoustic channels are discussed as they relate to digital communications, specifically with a moving AUV in the presence of load co-channel noise sources. WHOI's Utility Acoustic Modem (UAM) is introduced, and its transmission and reception algorithms are presented.

Using the Layered Control Architecture for Supervisory Control

A short description of the history of the idea of layered control and the MIT AUV Lab's implementation of the layered control architecture are presented. A method for framing operator commands in the context of the behavior-based layered control architecture is proposed.

SITE '99 Field Experiment Description

Field experiments to test a supervisory control protocol using a hybrid acoustic/RF communication network in Monterey Bay are described. Experiment infrastructure is outlined, including such resources as a Shore Control Computer (SCC), a Shore Node Computer (SNC), Data Repeater Moorings, Utility Acoustic Modems, a Support Ship, and the Odyssey IIb AUV Amphitrite.

Experiment Results

The results of field experiments in Monterey Bay are presented and analyzed. The *Message Detection Ratio* is examined in the context of the relative position of the source (the AUV) and the receiver (the data repeater moorings). Contributing factors to the *Data Transfer Rate* are examined, including multipath effects and UAM duty cycle. The *Byte-Error Ratio* is presented. Anecdotal evidence showing the effectiveness of the supervisory control capability is presented with examples from specific missions.

Conclusions and Future Work

The effectiveness of this work is examined, and suggestions are made for needed improvements and advancements. Possible applications to other areas of research are presented.

Chapter 2

The Challenges of Acoustic Communications

2.1 Introduction

In implementing a successful supervisory control capability for the Odyssey II AUV, one must recognize and deal with the constricting realities of the underwater acoustic environment. Low data rate, low signal-to-noise ratio, high propagation latencies, multiple propagation paths from source to receiver, uncertain connection reliability, and transmission errors all limit the effectiveness of acoustic communications, and thus will strongly shape the design of the supervisory control scheme. In this chapter, we discuss the challenges of underwater acoustic communications and describe a method for robust transmission and reception of acoustic messages in an unfriendly acoustic environment.

2.2 Modulation Techniques

Digital communication is, at its most basic, the creation and interpretation of two distinct signals representing binary digits – "one" and "zero". The typical method of transmitting digital information is by modulating the frequency of an analog signal. In the simplest method, known as Binary Frequency Shift Keying (BFSK), an "alphabet" is formed by associating tones at frequency f_1 Hz with binary 0 and tones at frequency f_2 Hz with binary 1. Using a BFSK modulation scheme, the string of bits *0111001010000111* would be represented by the sequence of tones $f_1 f_2 f_2 f_2 f_1 f_1 f_2 f_1 f_2 f_1 f_1 f_1 f_1 f_2 f_2 f_2$. The maximum data rate achievable with this modulation technique is directly related to the required length of each tone. In general, the required tone length depends on the frequency difference between tones, the SNR at the receiver, the estimator algorithm used by the receiver to determine the primary frequency component in the tone, and the Doppler shift in the perceived frequency at the receiver due to vehicle motion. A good starting place is to require that each tone be at least as long as the reciprocal of the frequency difference between tones. If we used 32 tones in the 12-16 kHz frequency band, each tone would have a bandwidth of 125 Hz, yielding minimum tone lengths of 8 milliseconds. Lower SNR and higher Doppler shift require an even more conservative number – 10 to 20 millisecond tone lengths are common, yielding a raw data rate of 50-100 bits per second with a BFSK modulation scheme.

A Quaternary FSK scheme involves transmitting frequencies representing bit-pairs, or *symbols*. The alphabet for QFSK modulation uses four symbols, as opposed to two for the method above; each symbol is assigned to a distinct frequency, f_1-f_4 (see Table 2.1). Using the QFSK alphabet, the string of bits *0111001010000111* would be represented by the sequence of tones $f_2 f_4 f_1 f_3 f_3 f_1 f_2 f_4$. QFSK modulation has the advantage of doubling the bit rate (sending two bits for a single tone) but the disadvantage of requiring twice as many frequencies. Following the previous example once more, a 10 millisecond tone length would yield a raw data rate of 200 bits per second (before application of any encoding techniques or other modifications). If desired, the modulation technique could be extended to group successively larger strings of bits, increasing bandwidth at the further expense of bandwidth efficiency.

Frequency	Bit-Pair
f_1	00
f_2	01
f_3	10
f_4	11

Table 2.1: A sample QFSK modulation alphabet.

2.3 Signal-to-Noise Ratio

In order to maintain an effective and reliable communication link, the controlled vehicle must transmit with enough power to maintain an adequate signal-to-noise ratio (SNR) at the receiver, and vice-versa. The primary factors that must be addressed when discussing SNR at the receiver are source level, loss due to *geometric spreading*, and ambient noise level.

The importance of source level to SNR is relatively simple; the more powerful the source, the higher the received level, and the higher the SNR. The source level may not be increased without bound, however – physical limitations of the transducer material and the phenomenon of cavitation (air bubbles may be created by overpowerful acoustic sources) impose an upper bound on this solution.

The term *geometric spreading* refers to the simple fact that acoustic waves propagate from a source in geometrically-defined patterns. Imagine an acoustic wave that originates as an explosive pulse from a point source in the ocean. The wave front will expand in all directions, forming a spherically-spreading wave. In an infinite medium, the wave front would propagate forever with the energy contained in the initial pulse spreading evenly over an ever-expanding spherical shell. Because of this spreading effect, the sound pressure at a point anywhere on the wave front is inversely proportional to the square of the distance from the source, $P \propto \frac{1}{r^2}$. While not a true loss mechanism (the original acoustic energy is still present in the shell, its point intensity is just decreasing), the effect at a more distant receiver is the same - a lower SNR.

Finally, ambient noise level can often be as loud or louder than the received acoustic signal, lowering the SNR. Ambient noise sources can be natural (crustaceans, surface waves, seismic), incidental man-made (shipping/fishing traffic), or vehicle selfnoise (LBL transducer, motor/prop vibrations).

2.4 Propagation Latencies

Compared with the speed of light in air, sound waves plod along interminably; the average speed of sound in water is 1.5x10^3 m/s, while the average speed of light is 3x10^8 m/s. Clearly, underwater acoustic communication over any significant range will involve a substantial round-trip time delay because of propagation latency. This precludes the implementation of any control scheme that requires the operator to provide continuous low-level command inputs to the AUV - by the time the operator reacts to sensor information he receives over the acoustic link, the information is five to seven seconds old! To make matters even more confusing, the time delay varies, depending on the distance between the AUV and its acoustic receiver.

To more capably control a vehicle that is seven seconds removed from him, the operator would prefer to use a supervisory control implementation that incorporates an inner and an outer feedback loop (see Figure 2-1). In this scheme the AUV's computer closes the inner feedback loop, robustly and intelligently stabilizing the vehicle's dynamics and accomplishing pre-planned mission-level goals. The outer loop is closed by the human operator, who observes the progress of the mission and issues high-level commands as the situation demands. Returning to the example in Chapter 1, when the operator determined that the weather at the surface was quickly becoming too rough to recover the AUV without endangering the ship's crew, he would, under a supervisory control implementation, send down a command cutting short the mission and directing the vehicle to rise to the surface. This type of high-level/low-level control interaction provides a measure of flexibility and agility that would otherwise be impossible.

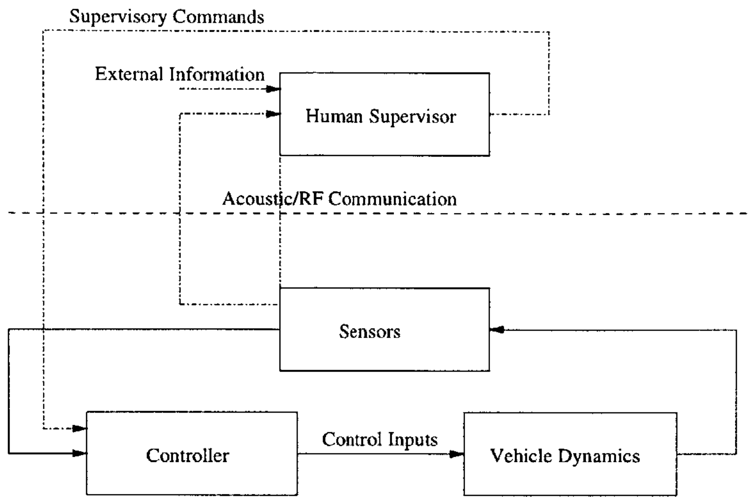


Figure 2-1: Block diagram of inner and outer control loops.

2.5 Multipath Propagation

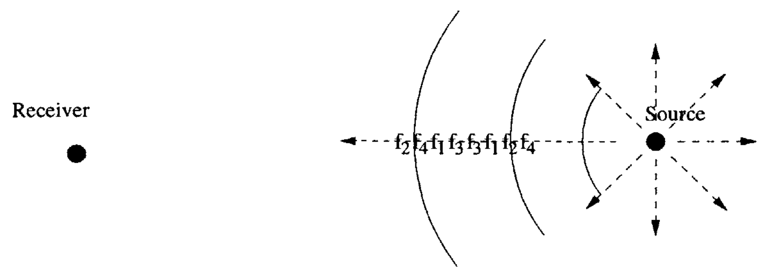
The simple digital modulation techniques described in Section 2.2 implied a reception algorithm that assumes that the receiver hears the packet of symbol tones ($f_2 f_4 f_1 f_3 f_3 f_1 f_2 f_4$) in the sequence generated by the transmitter, allowing a faithful reproduction of the original bit stream. This assumption causes no trouble if the ocean environment looks like the infinite medium of Figure 2-2(a) because the receiver only hears sound that travels on the straight-line path between the source and the receiver - the rest of the sound energy is radiated away in other directions.

The situation is more complicated if the environment looks like the bounded medium of Figure 2-2(b). In this case, the air-sea interface reflects rising sound rays back down into the ocean. Now, there exists a second path to connect the transmitter and the receiver - the surface bounce! This is an example of *multipath propagation.* Because the surface bounce path is longer than the direct path, a bounced packet arrives at the receiver delayed by $T = \frac{r_{direct}}{1500}[\frac{1-\cos\theta_B}{\cos\theta_B}]$ seconds compared to a direct packet, and the receiver hears two instances of the same transmission. The number of possible bounce paths increases with the introduction of additional reflective barriers: the sea floor, rock outcroppings, et cetera.

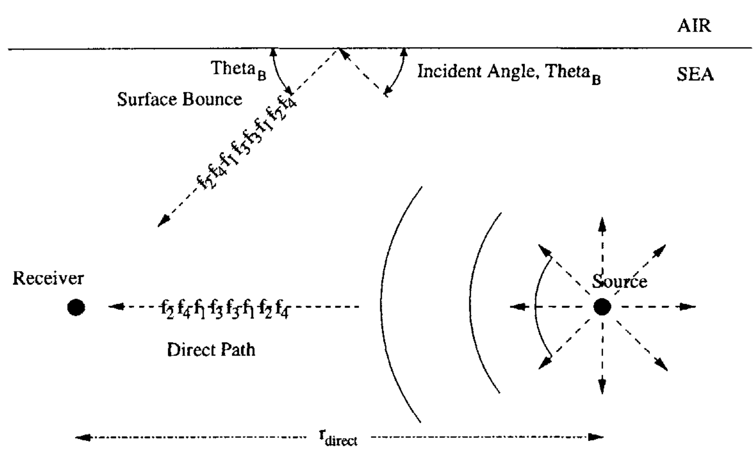
The significance of *multipath propagation* for digital communications is that the receiver's translation algorithm cannot be written assuming that the sequence of tones it hears is the sequence that was transmitted by the source. The delayed arrival of the bounced packet can cause two tones to arrive at the receiver at the same time – the receiver has no way of knowing which of the two tones is the real data and which should be discarded as an echo. However, the situation may be improved somewhat by modifying the way the source transmits symbols. If the source waits to transmit the next tone until all the echoes from the current tone die away, then the receiver can be guaranteed that all tones it receives in a single time frame correspond to direct-paths and bounce-paths of a single tone symbol. The receiver algorithm can again reliably decode the incoming signal – unfortunately, the communication rate is dramatically slashed due to the necessity to wait for echoes to die away. In the open ocean, the typical multipath echo duration is approximately 10 milliseconds, but is close to 100 or 200 milliseconds for acoustically-complicated ocean environments like the underwater canyons of Monterey Bay. The data rate can easily drop ten-fold if only this solution is employed. Other methods of combating the effects of multipath propagation will be discussed in greater detail in Section 2.7.

2.6 Communication Link Reliability

As in radio communications, environmental factors can have a significant effect on the reliability of underwater acoustic communications. In static situations (in which the transmitter and receiver are unmoving), once operators establish a link it is likely that it will continue to perform well, short of drastic changes in the environment or equipment. However, when either the receiver or the transmitter are moving, as is the case with an AUV, variations in the sound channel properties can make maintaining a reliable link very challenging. As the AUV strays further and further from the receiver, the strength of its signal fades at the receiver, making it more likely that ambient noise will swamp it. As the AUV travels around its underwater environs, local rock formations or thermal stratifications may interrupt the straight-line propagation path from the AUV to its receiver, or introduce reflections. The overall reliability of the

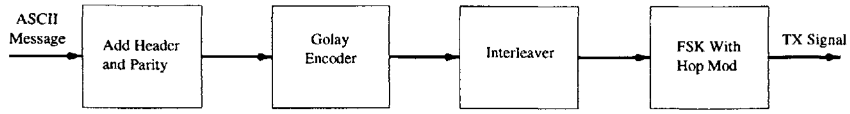


(a) Infinite medium – only one path exists between source and receiver.

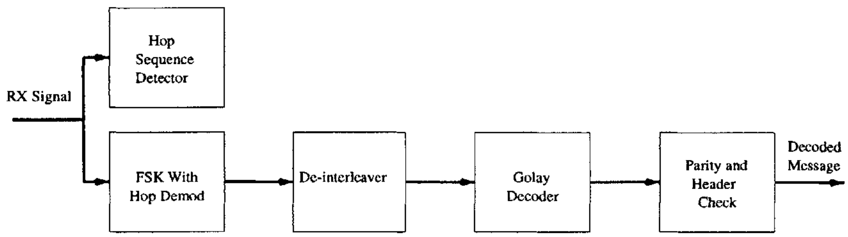


(b) Semi-infinite medium – two paths exist between source and receiver.

Figure 2-2: Multipath propagation.



(a) Transmit Algorithm



(b) Receive Algorithm

Figure 2-3: Block diagram of transmit and receive signal algorithms for the UAM, reproduced from ref. 17.

communication link is strongly dependent on both the complexity of the bathymetry[1] and on the vertical sound-speed profile[2], but it is almost inevitable that the link will fail at just the wrong time (don't forget Murphy's Law). The important thing to understand is that lapses in communication *will* occur - the success or failure of the mission depends on how gracefully one deals with these lapses.

2.7 WHOI's Algorithms for Robust Transmission and Reception

A highly robust signaling method was developed by Dr. Mark Johnson and others at WHOI for transmission and detection of acoustic messages in acoustically-challenging environments [17]. While the descriptions may at first seem overly detailed, a good understanding of the transmission and reception algorithms as well as the factors driving their design will provide a better understanding of bandwidth and range lim-

[1]Underwater canyons and "mountainous" terrain will create areas without a line-of-sight connection between the source and the receiver.

[2]Snell's Law says that a sound ray will bend toward depths with a slower speed of sound, possibly causing a formerly reliable line-of-sight path to fail.

Word	Bit 8	7	6	5	4	3	2	1	0
1	parity	packet number (8 bits)							
2	parity	source node i.d. (4 bits)				destination node i.d. (4 bits)			
3	parity	receive packet number (8 bits)							
4	parity	last source node i.d. (4 bits)				number of bad bytes (4 bits)			

Table 2.2: UAM Header Structure.

itations, as well as operational use. These algorithms deal effectively with low SNR, long multipath echo durations, and offer good error-detection and error-correction. These algorithms have been implemented and successfully exercised on WHOI's utility acoustic modem (UAM) during field experiments in Massachusetts Bay and Monterey Bay (and elsewhere).

2.7.1 Transmission Algorithm

Each message to be transmitted by the UAM may contain a maximum of 20 bytes (in this implementation), but the packet of bits that is actually transmitted is more than twice as long due to encoding and header data. The block diagram of the transmitter algorithm is shown in Figure 2-3(a).

Only standard ASCII characters are allowed in the packets that the UAM transmits – therefore, the message is re-formatted into 7-bit ASCII format, then truncated or padded with zeros to 20 characters. The 7-bit characters are made 8-bits bytes by adding an odd parity bit. Next, a 36-bit header (with fields as shown in Table 2.2) is pre-pended to the 20 byte message. The header contains an 8-bit number that acts as a packet identifier (0-255), a unique 4-bit identifier (1-15) for the source UAM, a unique 4-bit identifier for the UAM for whom the message is intended, the identifier of the last packet received (0-255), the unique 4-bit identifier of the last packet's source, and the number of byte-errors in the last packet. A destination node of zero indicates that the message should be decoded by any UAM that detects it. The fields pertaining to the last received packet (words 3 and 4 in Table 2.2) can help a receiving UAM decide if it should re-send a particular message, or if the source UAM did, in fact, receive the last message.

After reformatting and adding the header, the complete data packet occupies 196 bits. To add the ability to correct errors in the received data, the packet is encoded with a $\{23, 12\}$ Golay convolutional error-correction code, allowing the receiver to correct up to 3 bit-errors per code word. To use the code, the 196-bit packet first must be reformatted as 17 12-bit words (padding with 8 extra bits). The resulting data consists of 23 coded bits for each 12-bit input, or 17 23-bit words. This approach offers a measure of protection against bit-errors, but the algorithm is still vulnerable to long strings of bit-errors. While no further effort was made to protect against long strings of bit-errors, the coded words could have been systematically interleaved so that the least-significant-bits of each of the 17 words become the first 17 bits of the first word and so on, until all bits have been re-ordered. By interleaving the words, each bit in a string of consecutive errors would fall in a different code word, so that as many as 51 consecutive bit errors could be corrected (in this case). That is, if there were fewer than 52 consecutive bit errors, each original code word would have three bit-errors or fewer and therefore would be correctable using the $\{23, 12\}$ Golay encoding scheme.

Finally, a null bit is added to the coded, interleaved packet of 391 bits, and the packet is re-formatted as 196 2-bit symbols for transmission using the QFSK scheme. The QFSK modulation scheme assigns each pair of bits to one of four frequencies in a tone alphabet. The tone alphabet is changed for each pair of bits according to a pre-calculated frequency "hop" table to provide immunity from the effects of multipath propagation described in Section 2.5. Since transmission from one UAM to another is usually not synchronized, a method was devised to facilitate detection and synchronization at the receiver. At the beginning of each packet (header+message data), the source UAM plays a known sequence of 15 tones, each of about 8 milliseconds duration. After transmitting the preamble, the UAM waits 270 milliseconds before transmitting the packet to allow for echoes from the preamble to fully attenuate.

2.7.2 Receiver Algorithm

The block diagram of the receiver algorithm is shown in Figure 2-3(b). For most of the time, the detector is the only part of the receiver in operation, continually examining the received signal for the presence of the preamble sequence. The detector samples the received signal at 80 kHz and then produces a complex baseband sequence with a carrier frequency of 14.5 kHz, band-limited to +/-2.5 kHz. The detector separates the baseband signal into 20 discrete frequency bands, and computes the energy for each band at a rate of 400 samples per second. A matched filter is used to compare the energy in each band against the known preamble sequence.

Once the detector identifies the preamble sequence in the baseband signal, a portion of the baseband signal is captured and copied into memory on the UAM. The time series now in memory on the UAM contains both the preamble sequence and the 392-bit data packet (header+message data). The receiver decodes the time series simply by using the known "hop" sequence and the resulting tone alphabet to decode the hopped QFSK signal. To ensure that the demodulation algorithm is effective, the location of the first symbol (bit-pair) in the time series of data must be known with a high certainty. This synchronization function is achieved by using a matched filter to locate the beginning of the preamble sequence. Once the arrival time of the first symbol in the preamble is identified, the time series can be split into "frames" of the same duration as a symbol. The end of the preamble sequence is identified as 15 frames from the first symbol, and the first symbol of the real data is 270 milliseconds later in the time series (to match the 270 millisecond pause between the preamble and the data packet). The symbols are then demodulated by applying four parallel matched filters to each "frame"(one for each base frequency in the tone alphabet). After each symbol is demodulated, the matched filters are changed to match the hopping sequence so that multipath echoes recorded in the time series may be ignored.

Next, the receiver uses a pre-computed syndrome table for the $\{23, 12\}$ Golay code to correct any bit-errors (as far as possible). The final step in the receiver algorithm is to perform a parity check on each byte: bytes that do not pass the parity check are discarded and replaced by the underscore character (_), and a count of parity errors is kept.

Chapter 3

Using the Layered Control Architecture for Supervisory Control

3.1 Introduction

The *layered control* architecture was first proposed as a method for controlling fully autonomous land robots by Rodney Brooks and colleagues at the MIT Artificial Intelligence Laboratory [6]. Brooks' original formulation was based on his perception of the reactive way that insects seemed to operate in their environment. When an ant crawling across a table top runs up against a fruit basket, it turns left or right to try to go around it. When it "smells" a sugary treat, it follows the scent until it reaches its target. One would be hard-pressed to argue successfully that ants possess any higher cognitive ability; yet they thrive, going about the business of feeding their queen. Brooks proposed that the way ants operate could be represented by a number of competing survival- and goal-oriented *behaviors*, simple cognitive units that generated competing requests to the ant's legs based on the inputs from its antennae and other sensors. He demonstrated that outwardly complicated activities could be achieved by arbitrating the outputs of many such low-level behaviors, building from simple foundations.

The low computational requirements of a layered control architecture and the incremental way that layered control programs (missions) can be assembled make it an attractive alternative for experimental roboticists who often are testing hardware as they build it [3]. The layered control architecture has been tested on many land robots and has been adapted successfully for use in underwater vehicles by the MIT Sea Grant AUV Laboratory [16]. The following two sections will describe the MIT Sea Grant AUV Laboratory's implementation of the layered control architecture and discuss its place in the framework of supervisory control.

3.2 Behaviors and Arbitration

The elementary unit of the layered control architecture is a *behavior.* At its simplest, a behavior is a software module that maps sensory input to AUV actuator commands. Each behavior has the responsibility of trying to accomplish a single task; for instance, a behavior may be responsible for keeping the vehicle shallower than a certain maximum depth. If the behavior senses that the vehicle is too deep, it generates commands to try to drive the vehicle upward. Each behavior's output is in the form of a setpoint – a desired heading, speed, and/or depth. The following is a list of several of the behaviors used on the Odyssey AUV's:

- mission_timer – used to monitor the elapsed mission time. If too much time passes, the vehicle will shut down and float to the surface.
- depth_envelope - used to keep the vehicle shallower than a maximum depth and shallower than a minimum depth.
- altitude_envelope – used to keep a vehicle from crashing into the bottom.
- ascend – used to bring the AUV back to the surface.
- descend – used to lower the AUV to operating depths.
- setpoint – used to set the AUV on a desired heading, depth, and speed.

- waypoint – used to direct the AUV to a specific (x,y) location.
- launch_one – used to drive the AUV below the surface at the start of a mission.

Taken singly, no behavior could direct the AUV to perform all of the functions necessary for a science mission, but taken together in a scripted mission, the AUV can be made to carry out outwardly complex missions. An example mission file is shown in Figure 3-1. A layered control mission may consist of many of the above behaviors, each of which generates its own commands for the vehicle. In order to handle the possibility of conflicting commands, a simple method of resolving command conflicts is employed. Each behavior is assigned a priority at the time the mission is created. The highest priority behavior has the option of overriding commands from lower priority behaviors, and thus has final control of the vehicle. Priority is assigned to each behavior based on the potential consequences of that behavior not accomplishing its task. With this in mind, behaviors that try to protect the vehicle from dangerous situations are given higher priority than those that try to achieve a more abstract goal (i.e. locate an object on the sea floor).

It is useful to categorize behaviors as either goal-oriented or survival-oriented. Goal-oriented behaviors usually are continuously generating vehicle commands in an effort to achieve an assigned goal - for instance, a goal-oriented behavior might try to direct the vehicle to an (x,y) waypoint. A survival-oriented behavior is one which generally does not generate commands until it senses that the vehicle is in a dangerous situation (for example, going too deep). For this reason, the survival-oriented behaviors are usually given a high priority so that they may override goal-oriented behaviors that threaten the vehicle's survival.

Behaviors can exist in a mission in different states. They can be connected to the layered control architecture such that they generate commands for the vehicle - this is referred to as the *active* state. They can exist in an *uninitialized* state in which they do not generate commands until such time as they become active (automatically, or by input from a user). When a goal-oriented behavior accomplishes its assigned task, it enters into a *complete* state. If a behavior encounters an unrecoverable error, it

```
Countdown   0
UseThruster 1
UseElevator 1
UseRudder   1
UseParosci  1
UseKVH_DGI  1
UseKVH_DGC  1
UseLBL   1
UseCondTemp   1
UseRDI   0
UseModem 1
MonitorModem   5
InitDataFileSize  2
  sensor:                   m_pos_n(m) 0
  sensor:                   m_pos_e(m) 0
  sensor:    uo_magnetic_variation(rad) 0.26529
 behavior: mission_timer 1
  b_arg:                       time(s) 300
 behavior: ascend 2
  b_arg:             rudder_angle(rad) 0.174533
  b_arg:             ascent_angle(rad) 0.523599
  b_arg:                  speed(m/s) 1.5
  b_arg:                  end_depth(m) 0.8
 behavior: depth_envelope 3
  b_arg:                  max_depth(m) 12
  b_arg:                  min_depth(m) 1
  b_arg:      depth_cutoff_active(bool) 1
  b_arg:               cutoff_depth(m) 15
 behavior: setpoint 4
  b_arg:                  heading(rad) 1.39626
  b_arg:                      depth(m) 8
  b_arg:                    speed(m/s) 1.5
  b_arg:                       time(s) 120
 behavior: setpoint 5
  b_arg:                  heading(rad) 4.53786
  b_arg:                      depth(m) 10
  b_arg:                    speed(m/s) 1.5
  b_arg:                       time(s) 120
 behavior: launch_one 6
  b_arg:                    speed(m/s) -1
  b_arg:           elevator_angle(rad) 0.174533
  b_arg:                      depth(m) 4
  b_arg:                    timeout(s) 120
 behavior: broadcast 7
  b_arg:                 start_time(s) 0
  b_arg:                   duration(s) 43200
  b_arg:                   xs_delay(s) 10
  b_arg:                   xs_cycle(s) 15
  b_arg:                  uam_delay(s) 10
  b_arg:                  uam_cycle(s) 1
```

Figure 3-1: Sample mission file, from mission A9925518.

may enter an *error* state.

Finally, rather than have the layered control (or planning) level send final, resolved commands directly to the vehicle's actuators, commands are sent to the vehicle's dynamic controller. The dynamic controller is a classical automatic PD controller which receives the setpoints from the layered control level and generates the actuator commands required to achieve the setpoint. In this way, the vehicle dynamics are decoupled from the layered control level. The planning level still retains control over vehicle trajectory, but doesn't have to worry about the complexity of determining proper actuator settings.

3.3 Modifying a Layered Control Mission

The operator interacts with the vehicle by making changes in the vehicle's layered control program, or mission. As can be seen in Figure 3-1, each behavior in the mission has one or more parameters that help define the state the behavior is trying to reach (or avoid, in the case of survival-oriented behaviors). By changing the value of these parameters, the operator can have a profound impact on the mission. For instance, by changing the *heading(rad)* parameter of one of the *setpoint* behaviors, the operator can turn the vehicle in any desired direction. Similarly, if the mission contained a *waypoint* behavior, the operator could direct the vehicle to new points of interest by passing down new values of the goal waypoint.

Chapter 4

Field Experiments in Monterey Bay – September, 1999

4.1 Introduction

In September of 1999, the MIT Sea Grant AUV Laboratory conducted joint field operations with the Woods Hole Oceanographic Institution, Scripps Institution of Oceanography, University of Washington's Applied Physics Laboratory, and the Monterey Bay Aquarium Research Institute in Monterey Bay, California. The month-long experiment was titled the Synaptic Internal Tide Experiment (SITE) and brought together many diverse resources to study the dynamics of tide-induced internal waves in Monterey Canyon.

4.2 Experiments in Supervisory Control Using Acoustic Communications

During the course of the SITE experiment, three days were set aside to conduct tests of a radio frequency/acoustic communication network that would link a human operator with a sub-surface AUV. The goal of these tests was to explore the utility of a bi-directional acoustic communication link for the purposes of supervising and con-

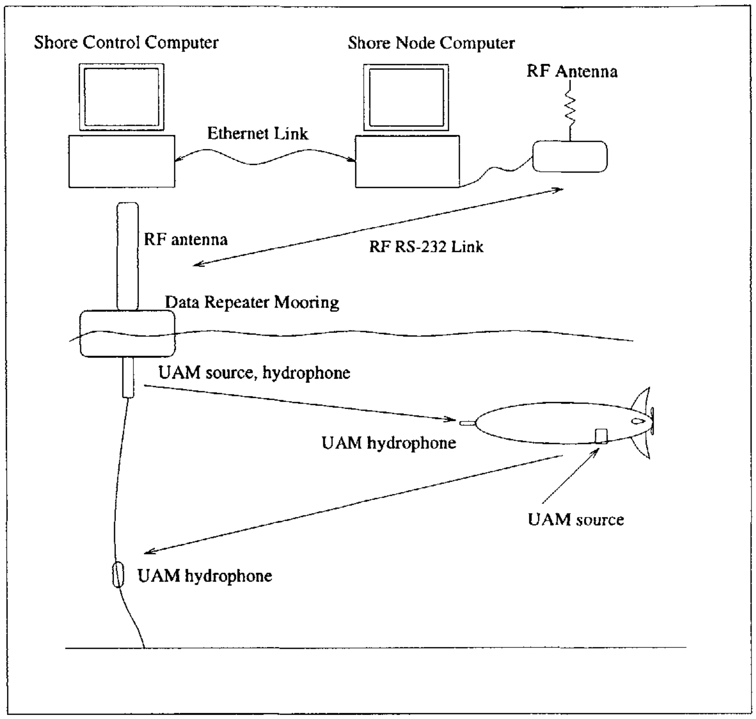


Figure 4-1: Overview of the RF/acoustic communication network implemented in Monterey Bay.

trolling an AUV from a remote location. The communication network is represented in Figure 4-1. The communication link allows the operator to do two things: monitor the vehicle's progress, and send commands to the vehicle to affect the course of the mission.

A hypothetical exchange between human operator and AUV would occur as follows (refer to Figure 4-1): the operator, stationed at the Shore Control Computer (SCC), dispatches an AUV-bound message to the Shore Node Computer (SNC) over an Ethernet network connection. The outgoing message is received by the SNC and transferred via radio frequency (RF) RS-232 serial link to one of three data repeater nodes - moorings situated in Monterey Bay (see Figure 4-2). The mooring receives the message, encrypts it via the algorithm described in Section 2.7, and broadcasts it acoustically into the surrounding ocean. When the AUV detects the acoustic signal, it records and decodes the message, then responds to the command/request it contains within. A reply by the AUV would follow the same path in reverse.

Nine vehicle missions were performed in which we gathered data on communication link performance and from which conclusions can be drawn about the value of the supervisory control capability.

Many resources were involved in conducting this experiment. These include the following:

- a shore-based control station, with two computers: the Shore Control Computer (SCC), and the Shore Node Computer (SNC). Each computer also had a FreeWave RF modem.
- three moorings acting as data repeater nodes, each having an Utility Acoustic Modem (UAM) and a FreeWave RF modem.
- an Odyssey IIb-class AUV, Amphitrite, outfitted with a UAM and a FreeWave RF modem.
- a support ship to tend the vehicle, with a Trackpoint acoustic tracking system, a FreeWave RF modem, and associated personnel.

4.3 Shore Control Station

The shore control station at MBARI consisted of one computer acting as the Shore Node Computer (SNC) and a second computer acting as the Shore Control Computer (SCC). Both computers were Dell OptiPlex GX1 models with Pentium II processors running RedHat Linux 6.0. The SNC's role was to act as the link to the data repeater nodes deployed in Monterey Bay. The SNC provided this link through a wireless RS-232 serial connection – a FreeWave RF serial modem. The location of the SNC was constrained by the maximum transmission range of the FreeWave RF transmitter – 10-20 km depending on antenna arrangement (see ref. 10). However, by using a local-area-network or a wide-area-network like the Internet, the SNC could act as a server for computers located in the same room, or around the globe. To demonstrate this capability, both the SNC and the SCC were connected to the Internet and installed in the same room at MBARI. The human operator used the SCC to communicate with the AUV by transferring all messages to and from the SNC using a standard FTP protocol. Note that although the FTP transfer was accomplished manually during the SITE experiment, the process of transferring data between the SNC and the SCC could easily be automated to enable the use of more sophisticated mission supervision software.

4.4 RF/Acoustic Data Repeater Moorings

Three moorings were provided by WHOI under sponsorship from the National Ocean Partnership Program (NOPP). These moorings acted as data repeater nodes in the communication network, bridging the air-sea interface by converting radio frequency transmissions to acoustic transmissions, and vice-versa. They provided a way for data to pass from the AUV's environment to a remote SNC located at the shore-based control station. Each mooring contained a Utility Acoustic Modem (UAM), a FreeWave radio frequency (RF) serial modem and an RF power controller. The FreeWave provided remote access to the UAM from the shore control station - it allowed data to be passed between the shore station and the mooring as if over a simple, hard-wired RS-232 serial connection. With appropriate setup, the FreeWave has a maximum range of 20 miles; however, in this experiment the range was limited by environmental conditions to about 10 km. The separate RF power controller allowed the human operator to turn the UAM and the FreeWave RF modem on and off to save battery power and to reset the mooring. The three moorings were placed in Monterey Bay at locations that would yield the greatest coverage area for communications in the area of operation, but remain within range of the Shore Control Station. In this report, the moorings will be referred to as M2, M3, and M4, as noted in Figure 4-2.

A source and two hydrophones were included on each mooring for transmitting and receiving acoustic signals. The source was a 3-ring 18 kHz Datasonics AT18DT; it was mounted on the base of the mooring housing (about 1.5 m below the water surface).

Mooring	Latitude	Longitude
M2	36°45.1 N	121°53.9 W
M3	36°47.6 N	121°51.0 W
M4	36°47.5 N	121°53.7 W

Table 4.1: Location of data repeater moorings in Monterey Bay.

The source had a toroidal beam pattern with a -3dB beam width of approximately +/- 30 degrees in the vertical plane and was omni-directional in the horizontal plane. The two hydrophones were Hi-Tech current-mode devices with a sensitive frequency range of 100 Hz to 30 kHz. One hydrophone was mounted on the lower endcap of the buoy and the other was resiliently mounted in a cage at various depths for moorings M2, M3, and M4. Both hydrophones were omni-directional [17].

4.4.1 WHOI's Utility Acoustic Modem (UAM)

The Utility Acoustic Modem (UAM) was also provided by WHOI. It consists of three circuit boards - a signal input board, a processor board, and a power amplifier board. The signal input board interfaces with the acoustic receiver, providing a low-noise 10 volt power supply for the hydrophones and filtering the received signal before passing it on to the analog-to-digital converter (ADC) on the processor board. The signal input board allows up to four channels of analog data to be filtered and amplified – for these experiments, the filters have a -3dB passband between 8 kHz and 20 kHz, and the gain of the amplifier is programmable via the processor board.

The processor board is based on a Texas Instruments TMS320c44 digital signal processor (DSP), with a clock speed of 60 megahertz. Onboard DC-DC converters allow the processor board to be powered by an external DC power supply with source voltage between 6 volts and 20 volts. The processor board uses a number of RS-232 serial ports and programmable input and output lines to communicate with external equipment, including the signal conditioning board and the power amplifier board. Onboard 12-bit analog-to-digital converters (ADCs) can be used to acquire up to four analog input channels simultaneously, with a combined sampling rate of 80 kHz. The DSP provides switching signals as input to the power amplifier board, which

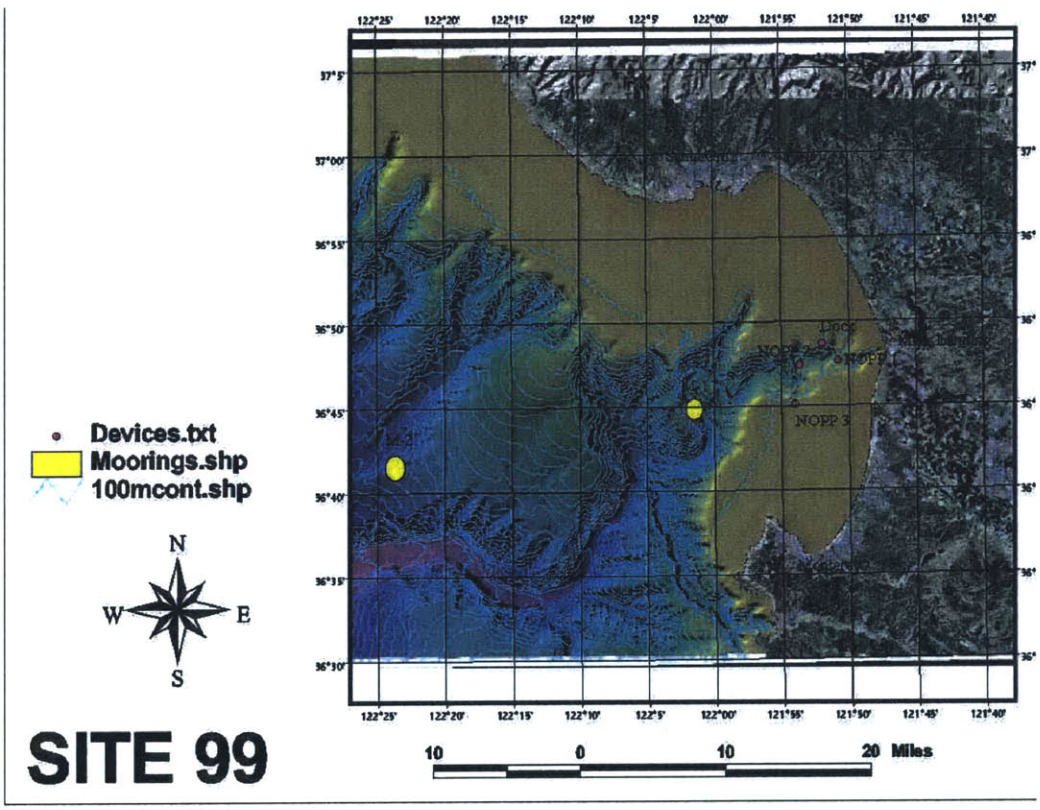


Figure 4-2: Locations of data repeater moorings M2, M3, and M4 in Monterey Bay (marked as NOPP 3,1, and 2).

subsequently drives the acoustic source.

The power amplifier board receives the switching signal from the DSP and uses a MOSFET switching technique to provide up to 50 watts of source power from a 6 - 20V power supply. With the Datasonics AT-18DT 3-ring sources installed in the moorings, the carrier frequency was 14.5 kHz with an operational bandwidth of +/- 2.5 kHz. The output power for the 3-ring source was approximately 170 dB re 1 μPa.

An operating system developed by researchers at WHOI named Acoustic Modem Software (AMS) is booted automatically from an EPROM on the processor board when external power is supplied. AMS provides a text-based interpreted programming language (similar to MATLAB) that can be used to write software for controlling UAM operation. AMS also offers a remote host interface that uses a server program running on a PC to provide access to UAM sub-systems through one of the RS-232 ports. Using the server program, all UAM sub-systems can be tested by running AMS scripts that reside on the PC, or AMS scripts can be downloaded to non-volatile memory on the processor board. When the UAM is powered up without the host server program in operation, AMS scripts in FLASH memory are run automatically.

4.4.2 Accessing the Repeater Moorings

When not being used, the data repeater moorings sat in a quiescent state, with the UAM powered down and the FreeWave in "sleep" mode. When the operator desired to talk or listen to the AUV, he would "wake up" the FreeWave by means of a Dual-Tone Multi-Frequency (DTMF) tone-dialer (from RadioShack) connected to a RF transmitter on shore. Once the FreeWave was "woken up", the operator would enter another DTMF tone-sequence to power up the mooring's UAM. Finally, the operator would start the remote host interface by executing the server program on the SNC. The SNC uses the RF RS-232 connection provided by the FreeWave modem to prompt the mooring's UAM to enter the host interface mode. In the host interface mode, the UAM has the ability to access the file-system of the SNC over the RF RS-232 serial connection, dramatically extending the range of the host interface access and allowing the mooring UAM to communicate with the SNC, located many kilometers

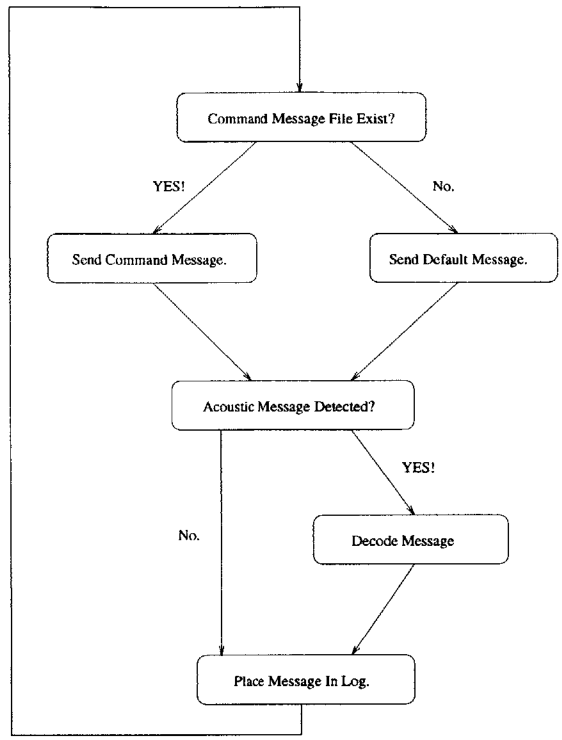


Figure 4-3: Logical organization of UAM program scripts for the data repeater moorings.

away.

Once the host interface link between the mooring and the SNC was achieved, the operator would command the UAM to execute program scripts that reside on the SNC. Written in AMS, these scripts contained the instructions that handle the encryption, transmission, detection, and decryption of acoustic messages. The scripts ran in a single thread of execution, cycling through the transmit-receive process about once every 40 seconds. The flow of operation is outlined in the flowchart in Figure 4-3).

When the operator wants to send a message to the AUV, he places the data in a special file on the SCC. To enable truly remote operation, the SCC and the SNC would share a network-mounted file-system – the SNC and the SCC would share a local directory via NFS, thus providing a line of communication between the UAM and the SCC, through the SNC. However, in this implementation, the transfer of files between the SNC and SCC occurred via manual use of standard FTP utilities.

```
UAM 223:74209 no AUV

UAM 223:74336 no AUV
UAM 223:74377 no AUV
UAM 223:74418 no AUV

UAM 223:74545 no AUV
A_D____q______ __P___
UAM 223:74627 no AUV
UAM 223:74668 no AUV
UAM 223:74709 no AUV
UAM 223:74750 no AUV
UAM 223:74791 no%__V
UAM 223:74832 no AUV
00120820000001490*35
00510070000031488*_5
00902490000041461*30
01292580000101320*34
01682580000091322*3b
02070730000081332*3a
02460760000071419*3b
02850750000081375*35
UAM 223:75185 no AUV
UAM 223:75226 no AUV
UAM 223:75267 no AUV
UAM 223:75308 no AUV
UAM 223:75349 no AUV
```

Figure 4-4: Log file of acoustic messages received at mooring M3. This log file is 25575326.M3.

During the transmission phase of operation, the UAM first checks the SCC to see whether this special file containing the operator's message exists. If it does exist, then the contents of that file are encoded and transmitted by the mooring UAM. If the file does not exist on the SCC, a default message is automatically transmitted in its place. After transmitting a message, the UAM samples the incoming acoustic signal for 15 seconds seeking encrypted data originating from the AUV. If it finds a valid transmission, the UAM decodes the message and saves it in memory. After thirty-one iterations of the listen-transmit sequence, the UAM writes a log file on the SCC containing all of the received messages. An example log file from mooring M3 is shown in Figure 4-4. This log file contains examples of each possible type of message: UAM default message (line 1 of text), false detect (line 6 of text), and AUV-generated (line 13 of text).

4.5 Vehicle Configuration

For the purposes of these tests the Odyssey IIb AUV, Amphitrite, was fitted with the standard suite of sensors and batteries[1]. Specialized equipment included a FreeWave RF serial modem for communicating with the SCC while on the surface[2], and a Utility Acoustic Modem (UAM) which would act as a transceiver for acoustic communications. Along with the UAM electronics, an omni-directional hydrophone for detecting transmissions, and a two-ring 18 kHz Datasonics source were installed in the vehicle. The source had a -3dB beam width of +/- 45 degrees in the vertical plane, with omni-directional sensitivity in the horizontal plane. The hydrophone was mounted in the free stream on the nose of the AUV and the transducer was mounted through the polyurethane hull on the aft underside of the vehicle (see Figure 4-1).

The vehicle's UAM operated in much the same way as the UAMs on the data repeater moorings. Instead of communicating with a remote computer over a FreeWave modem, the UAM is connected directly to a serial port on the AUV's Main Vehicle Computer (MVC). When the AUV is powered on, the UAM executes the AMS scripts that are saved in non-volatile memory on the processor board, which immediately begin running through the transmit/receive loop of Figure 4-3, regardless of whether or not a mission is running. If a mission is not running, the UAM will broadcast a default message - Figure 4-4 shows transmissions from the vehicle's UAM that were logged at data repeater mooring M3. Note the change in message content in the last third of the log file when the mission began; the message first changed from the UAM-generated default message to a message generated by the MVC – then, when the mission ended, it changed back to the default message.

[1]The AUV's altimeter was not functioning reliably – therefore it was disabled for these missions.

[2]Since the AUV has a very low profile on the surface, it was necessary to complete the link with the SCC by using a third FreeWave RF modem on the support ship as a repeater.

Figure 4-5: The AUV was deployed from the R/V Shana Rae, a fine vessel run by Captain Jim Christmann and Angie Christmann of Monterey Canyon Research Vessels, Inc.

4.6 Support Ship – the R/V Shana Rae

The AUV was launched from the R/V Shana Rae, a fifty-two-foot fishing vessel that has been converted for full-time use as a scientific research vessel (see Figure 4-5). A small A-frame on the stern offered a very effective launch and recovery platform. Engineers and scientists on board the Shana Rae provided support in case of emergencies or loss of RF communication with the vehicle on the surface. A Trackpoint acoustic tracking system kept the team aware of the vehicle's approximate location at all times, acting as a backup in case the acoustic communication system failed.

Chapter 5

Field Experiment Results and Data Set Descriptions

5.1 Introduction

We ran missions with the Odyssey IIb AUV named Amphitrite on Sunday, Monday, and Wednesday – 9/12/99, 9/13/99, and 9/15/99. These missions started out very cautiously, with the vehicle merely broadcasting a periodic message in order to test the communication network. In the next few missions, we tested the supervisory control capability by sending commands to the vehicle, though these missions were very limited in extent and duration in case of problems. Next, we tested the uni-directional maximum communication range by starting the vehicle near a node mooring and directing it to travel away from the mooring, again broadcasting the periodic message. Finally, we pushed the range of the system again, this time attempting bi-directional communication at the limits of its range. An example mission file can be found in Figure 3-1.

For each mission, the vehicle creates two log files: the first file contains a detailed log of all sensor values (polled every 200 milliseconds), and the second, called the syslog, contains a time-stamped log of higher-level events. For example, the syslog would include such high-level events as the start or conclusion of behaviors, and the reception of a command message. Nine of the missions resulted in useful data sets; these are listed in Table 5.1 along with general information about each mission. Several of the missions listed in this table were mainly focused on other scientific objectives, but the unobtrusive nature of the communications experiments meant that they could be performed alongside the primary work without disruption. In addition to the vehicle data files, Table 5.1 lists the associated log file(s) generated by the data node moorings. These log files contain a record of all detected messages that were sent by the AUV[1]. Due to an oversight by the author, messages sent by the human operator to the AUV were logged in only one instance (mission A9925518). A written record covers some of these instances, but this omission means that complete statistics about bi-directional communication performance may not be derived for every mission.

5.2 Communication Performance

In all nine of the missions, the AUV was broadcasting "heartbeat" information. The heartbeat contained the vehicle's elapsed mission time, heading, altitude, depth, and the local water temperature. Not only did this information give us an intimate picture of the course of the mission, but it also allowed us to collect broad statistics on the performance of the communication link. Here are some statistics over nine missions:

- The AUV sent 433 20-byte messages, for a total of 8,660 bytes.
- Of the 433 messages sent by the AUV, 265 were received by the data repeater moorings – a Message Detection Ratio (MDR) of 61.20%. This number was lower than expected, but is partly due to operator error – we neglected to log all messages that the data repeater moorings received.

[1]Several of the mooring log files cover periods of time during which the vehicle was in the water but no mission was running. However, the vehicle UAM broadcasts a default message in the absence of real vehicle data, so these files still contain valuable data. The filenames are: 25576607.M3, 25577856.M3, 25579105.M3, and 25867703.M3

Mission File	Start Time (hr:min:sec)	Duration (sec)	Latitude (deg., min. N)	Longitude (deg., min. W)	Mooring Log File
A9925518	20:45:51 Z	300.39	36°47.750	121°51.750	25575326.M3
A9925520	22:16:07 Z	411.80	vicinity of M3	vicinity of M3	25580346.M3 25581585.M3
A9925523	22:52:05 Z	483.00	36°47.509	121°51.653	25582807.M3 25584052.M3
A9925526	23:25:53 Z	1,229.00	vicinity of M3	vicinity of M3	25585254.M3 25600086.M3
A9925528	23:52:47 Z	1,570.00	vicinity of M3	vicinity of M3	25600086.M3 25601351.M3
A9925608	17:56:41 Z	10,052.60	36°46.986	121°56.004	25667391.M4 25668815.M4 25670155.M4 25657488.M2 25664193.M2
A9925811	16:46:00 Z	2,129.80	36°47.855	121°51.772	25862600.M3
A9925812	17:40:14 Z	2,394.40	36°47.840	121°51.787	25863862.M3 25865080.M3 25866379.M3
A9925813	19:01:30 Z	1,968.80	36°47.874	121°53.387	25868997.M3 25870449.M4 25871806.M4

Table 5.1: Vehicle mission information, with associated mooring log files.

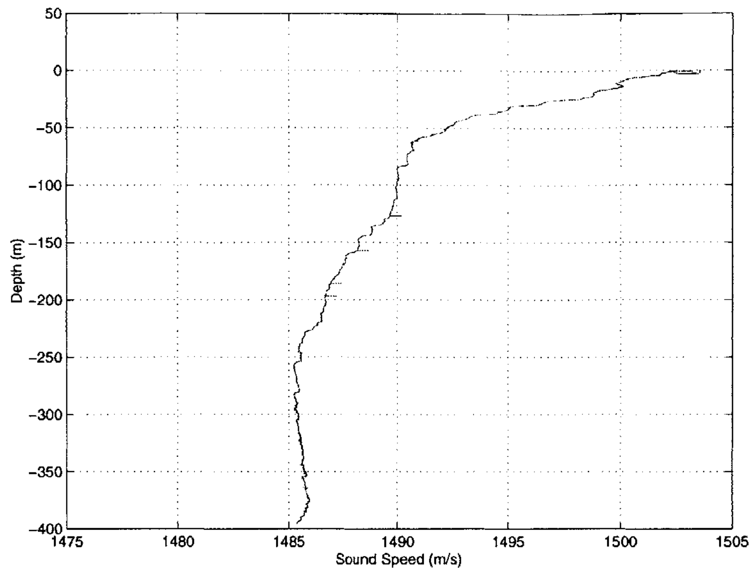
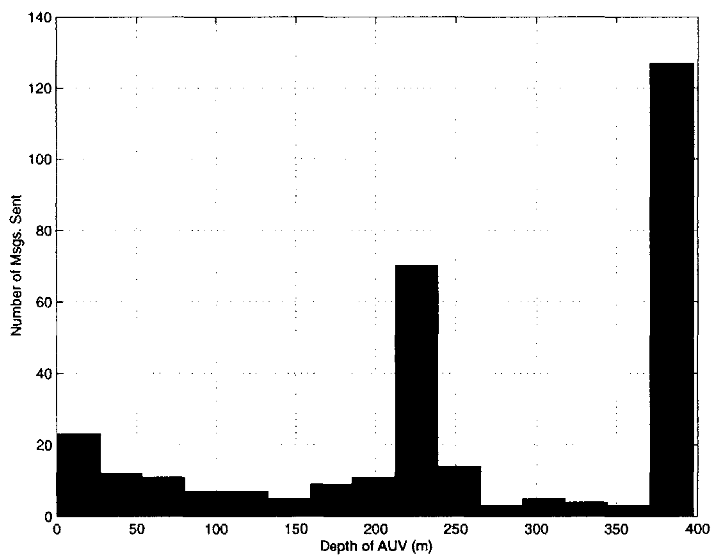


Figure 5-1: Sound speed profile from data taken during mission A9925608.

- Of the 5,300 bytes received by the moorings, 5,092 bytes were decoded correctly – a Byte-Error Ratio (BER) of 3.92%!
- Of the 208 incorrectly-decoded bytes, 143 bytes were successfully identified as errors while only 65 of the byte-errors continued unflagged.
- The average Data Transfer Rate (DTR) was 3.37 bits/second, or one 20-byte message every 47.4 seconds.

Figure 5-1 shows a sound-speed profile generated from data taken during mission A9925608. Note that the sound-speed gradient indicates a downward-refracting acoustic environment – i.e. sound rays will tend to bend toward the sea floor, reducing the sound power level at the receiver. Figure 5-2(a) and Figure 5-2(b) show histograms of the number of messages the AUV sent with respect to the two-dimensional distance between the AUV and the mooring and the depth of the AUV. Since the AUV sent these messages at regular intervals, the figures also show where the AUV spent most of its time underwater – within two kilometers of the moorings and at 230 meters and 400 meters depth.

(a) Histogram with respect to horizontal range from the mooring(s).



(b) Histogram with respect to depth.

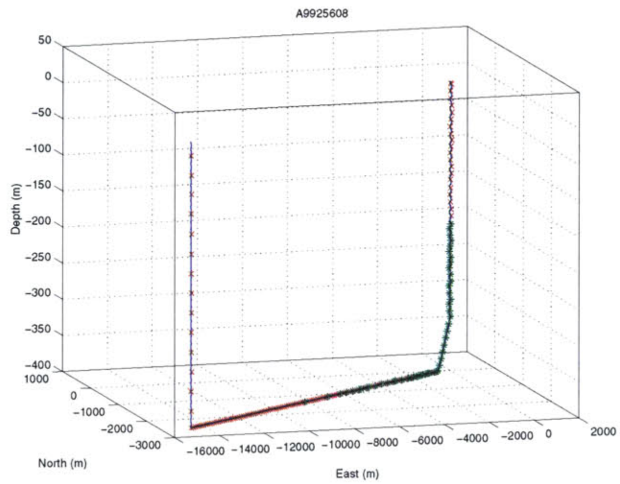
Figure 5-2: Histogram of messages sent by the AUV with respect to depth and horizontal range. These figures give an overview of where the AUV spent most of its time underwater.

5.2.1 Message Detection Ratio

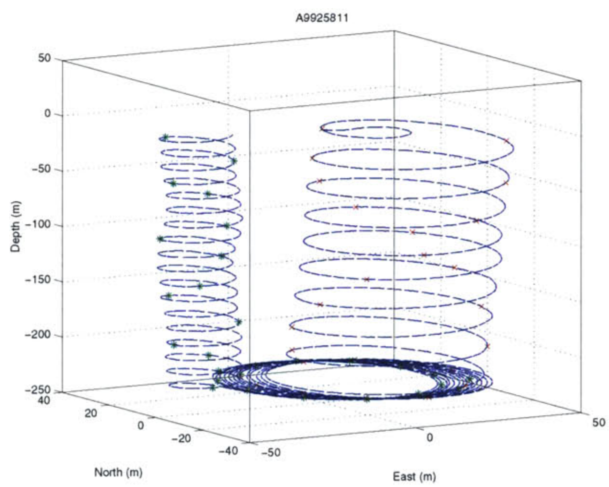
The overall message detection ratio (60.73%) at the moorings was poor, but the MDR must be viewed in the context of the type of message being transmitted. If the message carries vehicle position data, and is one of many such messages, then it may not be a disaster if a few messages are lost. However, if the importance of an individual message getting to the destination is high (for instance, if the operator is sending a new heading command to the AUV), then a missed message may mean the difference between (a) the AUV reversing course and (b) driving into a canyon wall.

If the overall MDR was so low, then we must ask, "Why?". A closer look at the communications statistics from each mission reveals that, for six of the nine missions, the MDR was above 90%, and above 87% for seven of the nine missions (see Appendix A for mission-specific communication statistics). Clearly, something anomalous happened with the remaining two missions. The two culprit missions are A9925608 and A9925811, which have a MDR of 27.47% and 53.70% respectively. Figure 5-3 shows the dead-reckoned trajectories of each mission with overlaid symbols indicating what the AUV's position was when it sent and received messages. Green asterisks (*) indicate messages that were successfully received by the data repeater mooring, and red "x"s indicate messages that were sent by the AUV but never received (that is, never logged in the mooring data files).

In mission A9925608, the AUV began its descent at 36°46.986 N and 121°56.004 W, about 3.6 kilometers south-of-west of mooring M4 (see Figure 4-2). From previous field deployments we expected this to be well within the useful range of the communications system (see report on communications experiments in Cape Cod Bay [17]). However, according to the logfile from mooring M4, the mooring didn't receive any of the AUV's messages until nearly twenty minutes had elapsed. Figure 5-3(a) shows the AUV's trajectory, beginning in the upper right-hand corner and ending in the lower left-hand corner. The figure clearly shows that the mooring log file recorded no receptions until the vehicle was 200 meters deep, at which point, the mooring suddenly began picking up the AUV's messages with almost 100% accuracy. This is



(a) Mission A9925608 had an MDR of 27.47%



(b) Mission A9925811 had an MDR of 53.70%

Figure 5-3: Two missions that had very low message detection rates. The trajectories are marked with green asterisks to indicate shore-bound messages that were successfully detected by the moorings and red X's to indicate missed messages.

strange, especially considering that the AUV's horizontal distance from the mooring had not changed significantly in the twenty minutes it took to reach 200 meters depth. A closer look at the mooring log file reveals that the very first entry is for a message sent by the AUV at 1,230 seconds (20 minutes) into the mission. The same pattern is seen in mission A9925811 – the mooring log file covering this mission begins with a message the AUV sent at 1,022 seconds into the mission.

These two facts point to a procedural problem on the mooring rather than a communication problem with the UAM's – in this case, the mooring log files and anecdotal evidence from the operator suggest that the operator neglected to command the mooring to automatically log the messages it detected. Based on the high detection ratio seen in both missions after the operator noticed his mistake and enabled logging, it can be safely assumed that most of the messages in the first 15-20 minutes of missions A9925608 and A9925811 *were* successfully detected but not logged, thus artificially lowering the average MDR. Continuing with this assumption, the re-calculated MDR's for missions A9925608 and A9925811 would increase to 39.01% and 100%, respectively! Mission A9925608 would still have a low MDR compared to the rest of the missions, and it will be shown that the majority of the missed messages can be attributed to steadily increasing distance from the moorings.

Range and Depth Effects

As expected, distance from the AUV to the mooring turned out to be an important factor in the MDR – the further the AUV is from the data repeater mooring, the less likely the mooring will hear its transmissions. Figure 5-4 shows that the MDR declined steadily with increasing distance from the moorings. The MDR was maintained above 80% out to about 4 kilometers, providing a reliable operational footprint of 50.26 square kilometers around each mooring. The furthest communication from the AUV to any mooring came in mission A9925608 – a transmission was successfully sent from the AUV to mooring M4, 8.36 kilometers away[2]. Communication was not reliable at

[2]This range is based on vehicle dead-reckoned information. The true number is almost certainly smaller, as the vehicle was traveling slower than expected due to trim problems.

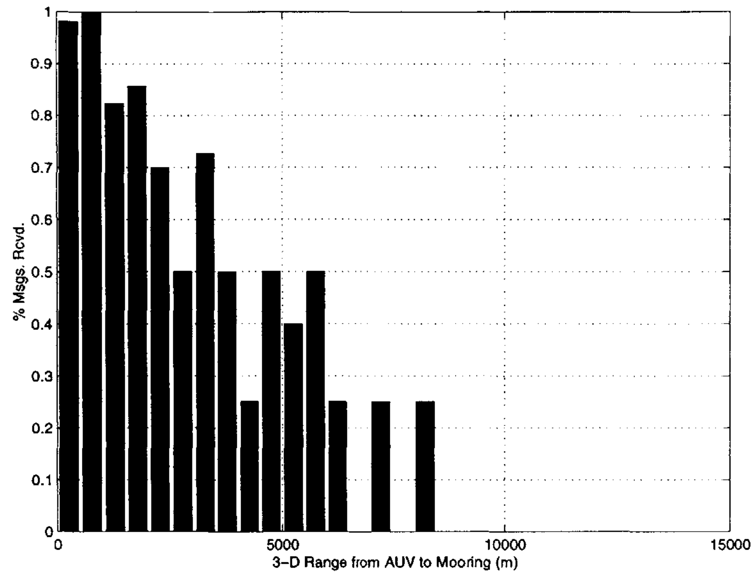


Figure 5-4: Relationship of 3-D range to MDR.

that distance, though, as the mooring was receiving about one out of every four messages the AUV sent. Beyond that distance, no more messages were received.

Clearly, the MDR depends on range from the mooring. But, does it also depend on the AUV's depth? Figure 5-5 attempts to answer that question by displaying a two-and-a-half-dimensional histogram relating horizontal distance from the mooring, the AUV's depth, and the MDR. While this figure is incomplete in the sense that no data was taken for the co-ordinates where white space exists, it can offer a glimpse at the co-dependence of depth, horizontal distance, and MDR[3]. At 230 meters, the MDR was greater than 70% out to a three-dimensional range of 3,500 meters[4]. However, at 400 meters depth, the MDR dropped off much more quickly – by the time the AUV was just 2,000 meters out, the MDR had already dropped to nearly 50%. The difference in MDR between the two depths may be a result of the sound speed profile (see Figure 5-1), but is more likely to be a result of the complicated topography –

[3]Figure 5-5 was created using data from only six of the nine missions. Missions A9925520, A9925526, and A9925528 were not included because the starting co-ordinates of the AUV were not recorded (and thus, the relative distance to the mooring could not be calculated). Also, the ambiguous data at the start of missions A9925608 and A9925811 was not included (a total of 36 data points).

[4]No more data was recorded at this depth - it is possible that the MDR would have remained above 70% further away from the mooring.

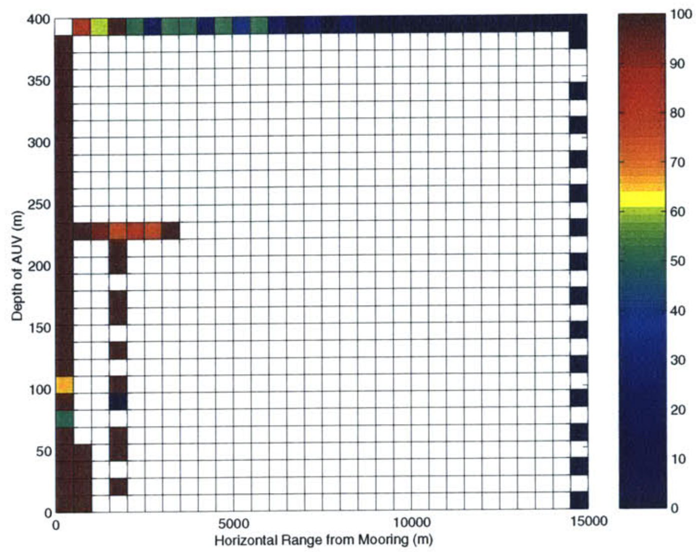


Figure 5-5: Two-dimensional histogram relating depth and range to MDR. White space indicates areas where no data exists.

the data at 400 meters was recorded when the AUV was running deep in Monterey Canyon. The AUV's transmissions lost too much energy in bouncing off rock walls, or were reflected in useless directions before they got to the data repeater moorings.

As a caveat, this figure obscures the fact that the data presented is a composite of six missions, each of which took place at different locations. In fact, local topography can have a dramatic effect on acoustic propagation; the causal relationship between topography and MDR is not included in this figure.

5.2.2 Byte-Error Ratio

The BER was chosen as a metric rather than the more common Bit-Error Ratio, due solely to the lack of information about bit-errors – the mooring logfiles only contained information about whole byte-errors. As such, some information may have been lost (a single byte-error can result from many combinations of bit-errors), but the Byte-Error Ratio still offers good insight into the performance of the communication system.

The BER of 3.92% averages out to less than one byte-error for each of the 265 re-

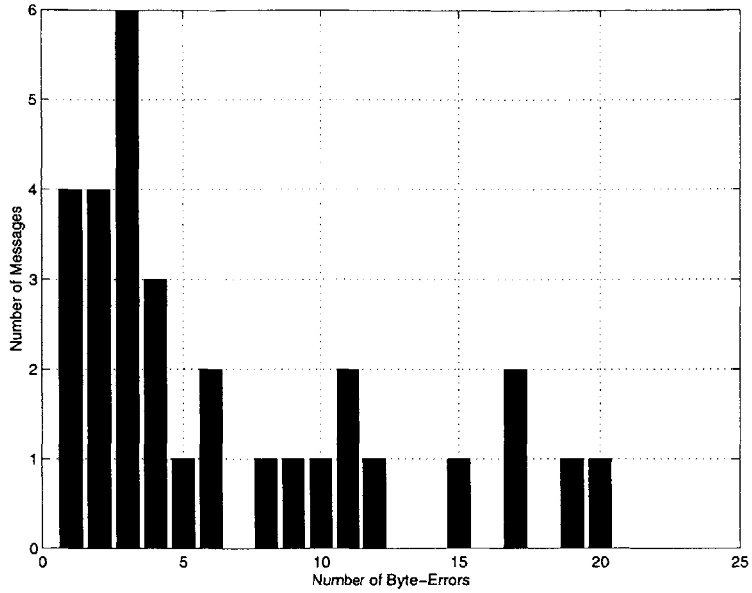


Figure 5-6: Distribution of byte-errors in each message. Messages with no errors were not included.

ceived messages. However,only 31 out of the 265 received messages actually contained errors, and each of these messages had an average of 6.7 byte-errors. The byte-error distribution is shown in Figure 5-6. It is clear that the Golay encoding algorithm was very effective in guaranteeing a high level of reliability in decryption. In fact, it might be useful to investigate other methods of encoding that are more bandwidth efficient – the Golay algorithm doubles the amount of data required to send a single message, approximately halving the communication rate. Finally, the parity bit on each byte of data was also helpful in identifying incorrectly decoded data – 143 out of the 208 byte-errors were correctly identified as errors.

5.2.3 Data Transfer Rate

The average DTR was very low, and fairly constant between missions, ranging from 2.90 bps to 4.26 bps[5]. The complete message included the 15-tone preamble, 270 millisecond channel-clearing delay, and 196 two-bit symbols. Each message took

[5]This number only includes the 20 bytes of "payload" data that is included in each message – it does not include the 36-bit header or the 187 added bits from the Golay encoding algorithm

approximately 4.3 seconds to transmit. In Monterey Bay, the primary limitations to the DTR was the presence of strong multipath echoes, and the power requirements of the UAM hardware.

Multipath Effects

As described in Section 2.7, the transmission and reception algorithms used by the UAM are based on Quaternary Frequency-Shift-Keying (QFSK). Due to the existence of multiple propagation paths from the source to the receiver (in the form of direct, bottom, surface, and canyon-wall bounces), the receiver may easily become confused if the transmitter is not properly configured. In order to counteract the effects of the echoes, the transmitter uses a pre-determined "hop sequence" in which the tone alphabet is continuously varied from symbol to symbol. Once the receiving UAM detects the preamble sequence in the message, it uses the known hop sequence to reliably reconstruct the transmitted message.

The maximum data rate was determined by how long multipath echoes were expected to last in the Monterey Canyon area (100-200 milliseconds), and thus, for how long each symbol (bit-pair) must be transmitted. Based on the available transmission bandwidth (the 12-17 kHz band was reserved for acoustic communications) the symbol length was set to 20 milliseconds. This limited the raw data rate to 100 bps[6]. For comparison, if the multipath duration was 10-20 milliseconds, the symbol length could have been shortened to five milliseconds, yielding a maximum data rate of 400 bps[7]. Half of the maximum bandwidth was spent on the encryption code, and 22.5% of the remaining bandwidth was used to transmit the header information – in Monterey, this left a maximum of 38.75 bps for the vehicle data. Power limitations would cut that number even further.

[6]Including the 15-tone preamble and channel-clearing delay, this drops to 92.1 bps.

[7]Remember, the minimum symbol length is determined by the lowest carrier frequency used – a good rule of thumb is to make the symbol long enough to ensure that it contains 10 or more wavelengths.

Power Limitations

During transmission, the UAM used an average of 10 W of power[8]. If the source UAM were to transmit at 100% duty cycle it would quickly drain the batteries, heating up and possibly destroying the electronics and/or the ceramic source itself. To conserve power and hardware, it was necessary to limit the transmission duty cycle to a more conservative value. In Monterey, the preamble, channel-clearing delay, and fully-coded packet took about 4.3 seconds to transmit, and (on average) the AUV sent one packet every 47 seconds, yielding an average duty cycle of 4.3:43.1, or 9.1%. The duty cycle could safely be increased to 4.3:24.37 (15%), and experts advise that, for safety, the duty cycle should not be pushed higher than 4.3:8.6 ($33\frac{1}{3}$%). With the 9.1% duty cycle used in Monterey, the effective DTR (for vehicle data) was limited to just 3.3 bps! During the remainder of the cycle (approximately 43 seconds), the UAM spends a maximum of 15 seconds listening for an incoming message packet and the rest doing computations and idling.

Another problem with the power system is revealed upon closer analysis of the heading data from mission A9925612, a long run out Monterey Canyon at a depth of 230 meters. As Figure 5-7 shows, there is a substantial periodic disturbance in the vehicle's measured heading! After looking at potential sources, the timing of the disturbances (they come about once every forty seconds) makes it clear that the perturbations are related in some way to the vehicle's UAM transmissions. It is hypothesized that the momentary high-current draw by the UAM transducer caused this heading variation. Obviously, this is undesirable, and the vehicle's power system design may need some more attention.

5.2.4 Networking Issues

In this network arrangement, the source and the receiver were in no way synchronized. They each ran through the cycle depicted in Figure 4-3 with no attempt at synchronization with the other. Thus, the chance existed that the two cycles could

[8]From personal correspondence with Peter Koski of WHOI.

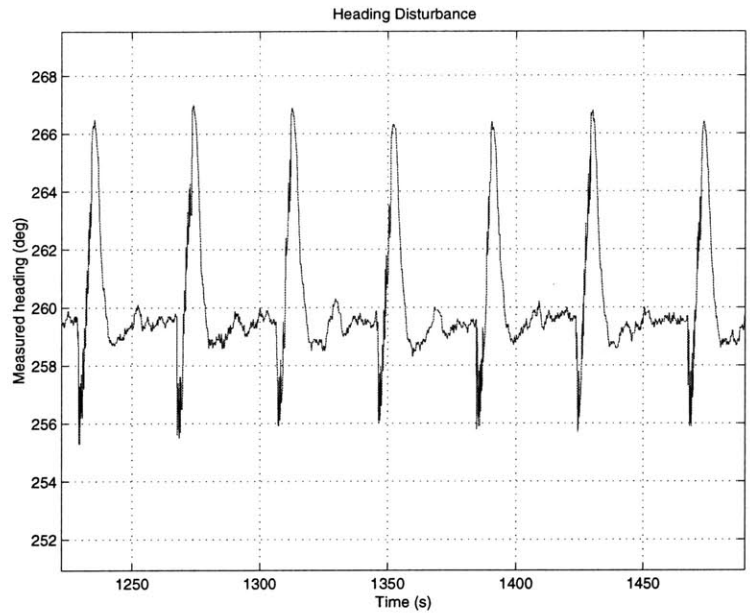


Figure 5-7: Measured heading from mission A9925528. Note the periodic heading disturbance, attributable to the power draw required during UAM transmissions.

align such that one UAM would be transmitting just as an incoming message arrived at its receiver. The incoming message would be completely obscured by the outgoing transmission. Even less optimal is the fact that the UAM is only listening for 15 seconds out of 47! There seems to be many opportunities for incoming messages to be missed completely.

5.3 Supervisory Control Performance

The primary goal of this work was to investigate the efficacy of a supervisory control capability, specifically, one made possible by use of an acoustic data link. This section covers examples of successful commands sent to the vehicle, which were subsequently executed, and lists pitfalls that were uncovered relating to operator error, DTR and propagation delay, and layered control behavior interactions.

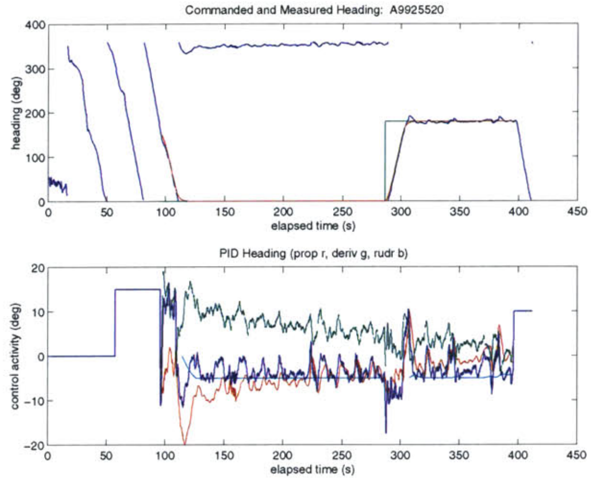
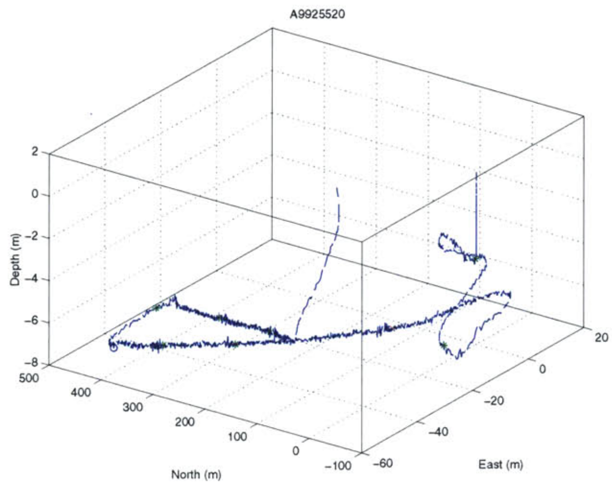


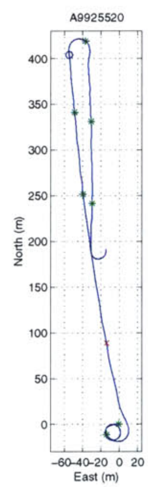
Figure 5-8: The upper plot shows the commanded heading in green and the estimated heading in blue. Note the step-change in commanded heading at 280 seconds.

5.3.1 Command Execution

The first mission in which we attempted to exercise the command capability was mission A9925520 (see Figures 5-8 and 5-9). This mission was designed such that the AUV would drive straight north at a depth of seven meters for 420 seconds before ascending to the surface. Approximately 280 seconds into the mission, we sent down a command to change the AUV's desired heading to 180 degrees, due south. The vehicle's event log has a record of the transmission being received, and the data file shows that the commanded heading changed as a result of the received acoustic command – note the step-change in the commanded heading in Figure 5-8. The vehicle's response to this change in commanded heading can be clearly seen in both plots of Figure 5-9 – on the plan view, the message reception is marked by a blue circle in the upper-left, followed shortly by a hard right turn that continues until the vehicle reaches a heading of 180° degrees. This clearly demonstrates the capability to interact with the vehicle while using an acoustic communication link. The key is that all the high-bandwidth control issues are dealt with on the vehicle – this leaves the operator with the responsibility to supervise the mission's progress and make high-level changes.



(a) An isometric view of mission A9925520.



(b) A plan view of mission A9925520.

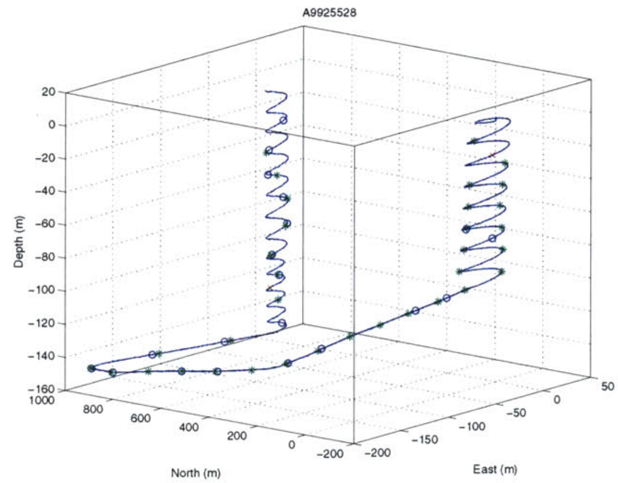
Figure 5-9: The first mission successfully demonstrating operator-control of the AUV (figures not to scale).

5.3.2 Operator Error

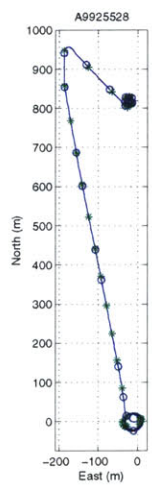
In mission A9925528, the plan was to do much the same as we had in mission A9925520 – send the AUV north, and then turn it, this time to a heading of 260° degrees, out the canyon. Unfortunately, in the heat of the moment, and in an attempt to compose the command and send it in the right transmission time window, I sent down the new heading in units of degrees, rather than the required units of radians! Of course a heading of 260 radians corresponds to 137.2° degrees, and so the vehicle took the course shown in Figure 5-10. This kind of error is only to be expected when testing new systems, but we were lucky that I hadn't made a different typo and accidentally commanded the vehicle to dive down and crash into the sea floor! This accident highlights the need for some sort of operator aid, probably in the form of a Graphical User Interface (GUI), that could help catch errors before they are passed on to the vehicle.

5.3.3 Operational Limitations Due to Low Data Transfer Rate and Propagation Latencies

The next mission was A9925526, a mission of approximately twenty minute duration. This mission was designed as a test to see if the new supervisory control capability would allow us to perform a mission in more challenging underwater terrain than we usually allowed – near one of the walls of Monterey Canyon. Monterey Canyon runs at about 260° and has rock walls that are hundreds of meters high – a very challenging operational area. The AUV does not carry any sonars for obstacle avoidance (save the altimeter), so we usually try to ensure that the planned missions keep the AUV well away from any rock walls. However, by correlating the surface ship's report of the AUV position (from the Trackpoint system) with the ship's GPS position, we could monitor how close the vehicle was to the northern canyon wall. We decided to direct the AUV to drive straight north toward the canyon wall, making sure to set its maximum mission time low enough that it couldn't reach the wall. However, if, during the mission, we felt that the AUV could still safely swim north, we could



(a) Isometric view of mission A9925528.



(b) Plan view of mission A9925528.

Figure 5-10: Mission showing the effects of operator carelessness. The operator had planned for the AUV to turn left to 260°, but used the wrong units in the command. Instead the AUV turned right to 137°.

periodically lengthen the mission time by using the supervisory link.

The vehicle was continuously tracked with the Trackpoint system and it was also broadcasting "heartbeat" data every forty seconds, containing the remaining number of seconds before the vehicle would rise to the surface and shutdown. With this information, the operator could make a decision to extend the mission or let the mission expire, based on how close to the canyon wall the vehicle was. Our first attempt at sending down a command to extend the second setpoint outlined the difficulties of working with large propagation latencies – since the messages come only once every forty-fifty seconds, and the message takes approximately 5 seconds to get to the AUV, we ended up missing the mission cutoff time. Our command to extend the mission time arrived approximately 2.5 minutes too late!

It became painfully clear that the bandwidth and propagation latencies of the communication link make it impossible for the operator to really *react* to information that he receives over the acoustic link. Instead, the operator must always be anticipating what will happen in a few minutes, making predictions, and generally ready to act before he learns of a problem. However, the acoustic link is still very useful for intervening in non-emergent situations and for obtaining up-to-date information about sensor readings, vehicle orientation and status, and mission status! None of these capabilities would exist without taking advantage of the limited capabilities of the acoustic link.

5.3.4 Unexpected Layered Control Behavior

Finally, the mission displayed in Figure 5-11 shows what can happen if the operator does not fully think out the consequences of changing a behavior parameter. In this case, the AUV was performing a long transect out the axis of Monterey Canyon, and the science goal dictated that the vehicle needed to stay within 100 meters of the bottom. In Monterey Canyon, the sea floor slopes away as one gets further from shore, so the AUV needed to descend as well to stay within 100 meters of the bottom. Unfortunately, the AUV's altimeter was non-functional, so it could not be relied on to keep the vehicle away from the bottom. As a compromise, it was decided to have

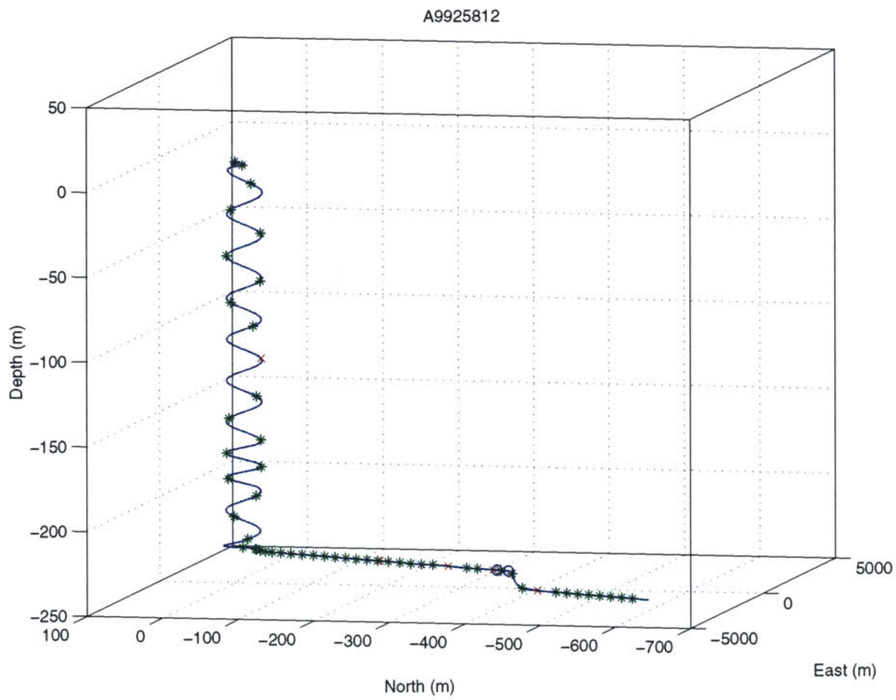


Figure 5-11: This mission was a long run out the axis of Monterey Canyon. Note the unexpected end of mission indicated by the absence of a return-to-surface trajectory.

the vehicle travel at a constant depth, but use the supervisory control capability to manually lower the vehicle by 10 meters about halfway through the mission.

Examining Figure 5-11 closely, one can clearly see the point at which the AUV received the message to dive deeper – the trajectory dropped by 10 meters. However, almost 200 meters later, the mission suddenly aborted, and the vehicle drifted to the surface. Post-analysis revealed that the *depth_envelope* behavior caused the abort when the vehicle ventured below a cutoff depth. This result makes sense with hindsight, but during the mission, it was far from clear what had caused the problem. This is an example of how the subtleties of behavior interaction can be problematic, especially when one starts modifying behavior parameters.

Chapter 6

Conclusions and Recommendations for Future Work

This thesis provided a unique opportunity to examine the usefulness of an acoustic communication link in enabling supervisory control for an Autonomous Underwater Vehicle. Acoustic communication is a challenging task due to the dynamic properties of the link: the locations of the source and/or the receiver are constantly changing; multiple propagation paths exist, making decryption of data a difficult task; range and bandwidth are limited by the physics of the medium; power availability is low due to space constraints on an AUV. However, given all these difficulties, it is still possible – data transfer links can be supported at up to 2,400 bits per second in uncomplicated, forgiving acoustic environments. Lower performance, but still workable communication systems can be maintained even in the toughest operating environments – the depths of Monterey Canyon, where multipath propagation is extremely strong.

Furthermore, a successful supervisory link with an AUV is possible with the low data rate link used in Monterey Bay. We demonstrated a remote, wireless command presence at the AUV from distances of 16 kilometers using a combination radio and acoustic communication network. We provided to operators, sitting at a desk on shore, the ability to monitor the AUV's heading, depth, altitude, and the measured water temperature with sub-minute granularity.

6.1 Recommendations for Future Work

During the course of this work, several possible avenues of future work were highlighted as necessary improvements to the current work, or as newly-enabled by the current work.

6.1.1 Information Displays

A noticeable lack of functionality in the current work is the absence of any kind of graphical display to make data coming back from the vehicle more readable. This is especially important if the operator is attempting to use the data to make informed decisions about how to change the course of the vehicle's mission! Too often, more time is spent interpreting textual versions of vehicle data, than is spent deciding what the vehicle should do next.

A more specialized type of information display is a vehicle dynamics prediction display. As mentioned before, the low message frequency and high propagation delay forced the operators to spend a lot of time trying to figure out what the vehicle was going to be doing five minutes in the future. Instead of the operator trying to predict those quantities, it would be better if the operator's computer were running a kind of state estimator (perhaps a Kalman filter) that was continually generating vehicle state predictions (position, velocity, orientation) based on "heartbeat" information obtained from the vehicle. This would be similar to the predictive displays used in in the bridges of super tankers and air craft carriers to aid the pilots of these massive vessels.

6.1.2 Vehicle-to-Vehicle Communication and Networking

As untethered, unmanned underwater vehicles become more reliable, robust, perhaps commercialized, systems, people will start to want to use more than one at a time. Already, many researchers in ground robotics have performed work suggesting how multiple robots could cooperate to accomplish a search task more effectively together than alone. The availability of a proven wireless underwater communication method could make inter-vehicle cooperation possible in the underwater arena. Vehicle's that are scanning the seabed for buried mines and building maps of their immediate vicinity could share those maps with other vehicles to make the task of autonomous searching and navigation easier. Or, vehicle's could use the communication link to achieve formation swimming, thus allowing multiple vehicles to cooperate in large-scale patterned searches. Finally, much work needs to be done to investigate how the current acoustic transmission and reception algorithms scale with the addition of more nodes in the network.

Appendix A

Tables

Mission name	A9925518
Mooring logfile	25575326.M3
Number of messages transmitted	8
Number of undetected messages	0
Message detection rate	100%
Number of bytes transmitted	160
Number of byte-errors	1
Number of undetected byte-errors	0
Data rate	4.26 bits/sec or one msg. per 37.5 sec

Table A.1: Communication Statistics for mission A9925518.

Mission name	A9925520
Mooring logfiles	25580346.M3 25581585.M3
Number of messages transmitted	8
Number of undetected messages	1
Message detection rate	87.5%
Number of bytes transmitted	160
Number of byte-errors	23
Number of undetected byte-errors	2
Data rate	3.11 bits/sec or one msg. per 51.5 sec

Table A.2: Communication Statistics for mission A9925520.

Mission name	A9925523
Mooring logfiles	25582807.M3 25584052.M3
Number of messages transmitted	12
Number of undetected messages	0
Message detection rate	100%
Number of bytes transmitted	240
Number of byte-errors	15
Number of undetected byte-errors	3
Data rate	3.98 bits/sec or one msg. per 40.25 sec

Table A.3: Communication Statistics for mission A9925523.

Mission name	A9925526
Mooring logfiles	25585254.M3 25600086.M3
Number of messages transmitted	30
Number of undetected messages	1
Message detection rate	96.67%
Number of bytes transmitted	600
Number of byte-errors	26
Number of undetected byte-errors	3
Data rate	3.91 bits/sec or one msg. per 41.0 sec

Table A.4: Communication Statistics for mission A9925526.

Mission name	A9925528
Mooring logfiles	25600086.M3 25601351.M3
Number of messages transmitted	36
Number of undetected messages	2
Message detection rate	94.44%
Number of bytes transmitted	720
Number of byte-errors	57
Number of undetected byte-errors	6
Data rate	3.67 bits/sec or one msg. per 43.6 sec

Table A.5: Communication Statistics for mission A9925528.

Mission name	A9925608
Mooring logfiles	25667391.M4 25668815.M4 25670155.M4 25657488.M2 25664193.M2
Number of messages transmitted	182
Number of undetected messages	130
Message detection rate	28.57%
Number of bytes transmitted	3,640
Number of byte-errors	2,634
Number of undetected byte-errors	14
Data rate	2.90 bits/sec or one msg. per 55.2 sec

Table A.6: Communication Statistics for mission A9925608.

Mission name	A9925811
Mooring logfile	25862600.M3
Number of messages transmitted	54
Number of undetected messages	25
Message detection rate	53.70%
Number of bytes transmitted	1,080
Number of byte-errors	521
Number of undetected byte-errors	7
Data rate	4.06 bits/sec or one msg. per 39.4 sec

Table A.7: Communication Statistics for mission A9925811.

Mission name	A9925812
Mooring logfiles	25863862.M3 25865080.M3 25866379.M3
Number of messages transmitted	57
Number of undetected messages	5
Message detection rate	91.23%
Number of bytes transmitted	1,140
Number of byte-errors	137
Number of undetected byte-errors	14
Data rate	3.81 bits/sec or one msg. per 42 sec

Table A.8: Communication Statistics for mission A9925812.

Mission name	A9925813
Mooring logfiles	25868997.M3 25870449.M4 25871806.M4
Number of messages transmitted	46
Number of undetected messages	4
Message detection rate	91.30%
Number of bytes transmitted	920
Number of byte-errors	154
Number of undetected byte-errors	16
Data rate	3.74 bits/sec or one msg. per 42.8 sec

Table A.9: Communication Statistics for mission A9925813.

Appendix B

List of Acronyms

ABE	Autonomous Benthic Explorer
ADC	Analog-to-Digital Converter
ASCII	American Standard Code for Information Interchange
AUV	Autonomous Underwater Vehicle
BER	Byte-Error Ratio
bps	bits per second
BFSK	Binary Frequency-Shift-Keying
DTMF	Dual-Tone Multi-Frequency
DSP	Digital Signal Processing
DTR	Data Transfer Rate
FTP	File Transfer Protocol
GUI	Graphical User Interface
MBARI	Monterey Bay Aquarium Research Institute
MFSK	Multiple Frequency Shift Keying
MDR	Message Detection Ratio at the mooring
MVC	Main Vehicle Computer
NOPP	National Ocean Partnership Program
PD	Proportional-Derivitave
PPM	Pulse Position Modulation

QFSK	Quaternary Frequency Shift Keying
REMUS	Remote Environmental Monitorint UnitS
ROV	Remotely-Operated Vehicle
RPC	Remote Procedure Call
SCC	Shore Control Computer
SNC	Shore Node Computer
SNR	Signal-to-Noise Ratio
UAM	Utility Acoustic Modem
WHOI	Woods Hole Oceanographic Institution

Bibliography

[1] James W. Bales, 1998. personal correspondence.

[2] James G. Bellingham, 1998. personal correspondence.

[3] James G. Bellingham and Douglas Humphrey. Using Layered Control for Supervisory Control of Underwater Vehicles. In *Intervention '90 - On The Shores Of Tomorrow*, pages 175–181. Marine Technology Society, June 1990.

[4] James G. Bellingham and John J. Leonard. Task configuration with layered control. Technical Report MITSG 94-24J, MIT Sea Grant Autonomous Underwater Vehicles Laboratory, 1994.

[5] David Brady and Josko A. Catipovic. Adaptive Multiuser Detection for Underwater Acoustic Channels. *IEEE Journal of Oceanic Engineering*, 19(2):158–165, 1994.

[6] Rodney Brooks. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2, 1986.

[7] Inc. Datasonics, 1999. http:/www.datasonics.com.

[8] Ira Dyer. Fundamentals and Applications of Underwater Sound.

[9] W. R. Ferrel and T. B. Sheridan. Supervisory control of remote manipulation. *IEEE Spectrum*, 4(10):81–88, 1967.

[10] FreeWave Technologies, Inc. *FreeWave Spread Spectrum Wireless Data Transciever User Manual*, v3.94 edition.

[11] L. Freitag, M. Johnson, and J. Preisig. Acoustic communications for UUV's. *Sea Technology*, 39(6):65+, 1998.

[12] Steven D. Gray, James C. Preisig, and David Brady. Multiuser Detection in a Horizontal Underwater Acoustic Channel Using Array Observations. *IEEE Journal of Oceanic Engineering*, 45(1):148–160, 1997.

[13] Robert Grieve, 1998. personal correspondence.

[14] Robert J. Grieve. Operations Post Cruise Report, Cruise #MIT98-01. Technical report, MIT Sea Grant AUV Laboratory, 1998.

[15] Robert J. Grieve. Operations Post Cruise Report, Cruise #MIT98-02. Technical report, MIT Sea Grant AUV Laboratory, 1998.

[16] Loch J., Waller E., Bellingham J. G., Beaton R., and Triantafyllou M. Software Development for the Autonomous Submersible Program at MIT Sea Grant and Draper Laboratory. In *Proceedings of the Sixth International Symposium on Unmanned Untethered Submersibles Technology*, pages 25–32, 1989.

[17] Mark Johnson. MURI/AOSN Project Cape Cod Bay Experiment: Cruise Report for the Acoustic Communications Effort. Technical report, Woods Hole Oceanographic Institution, 1998.

[18] Jon Kristensen and Karstein Vestgard. Hugin – An untethered underwater vehicle for seabed surveying. In *Proceedings of Oceans 1998*, 1998.

[19] Bradley A. Moran, 1998. personal correspondence.

[20] D. Porta. Underwater acoustic communications. *Sea Technology*, 39(2):49+, 1998.

[21] Carlos Sanchez, Peter Koski, and David Brady. Sequence Position Modulation for Surf-Zone Underwater Acoustic Communications. In *Eleventh International Symposium on Unmanned Untethered Submersible Technology*, number 11, pages 270–279. UUST, Autonomous Undersea Systems Institute, August 1999.

[22] Craig P. Sayers, Richard P. Paul, Louis L. Whitcomb, and Dana Y. Yoerger. Teleprogramming for subsea teleoperation using acoustic communication. *IEEE Journal of Oceanic Engineering*, 23(1):60–70, January 1998.

[23] Thomas B. Sheridan. *Telerobotics, Automation, and Human Supervisory Control.* The MIT Press, Cambridge, Massachusetts and London, England, 1992.

[24] R. Stokey, T. Austin, C. von Alt, M. Purcell, R. Goldsborough, N. Forrester, and B. Allen. AUV Bloopers or Why Murphy Must Have Been an Optimist: A Practical Look at Achieving Mission Level Reliability in an Autonomous Underwater Vehicle. In *Eleventh International Symposium on Unmanned Untethered Submersible Technology*, number 11, pages 32–40. UUST, Autonomous Undersea Systems Institute, August 1999.

[25] Roger Stokey, 1998. personal correspondence.

[26] Nils Storkersen, Jon Kristensen, Arne Indreeide, Jon Seim, and Tom Glancy. Hugin–uuv for seabed surveying. *Sea Technology*, 39(2):99–104, February 1998.

[27] Nils Storkerson, Henrich Henriksen, and Rolf Arne Klepaker. Acoustically controlled untethered underwater vehicle systems for deep water seabed mapping. In *Proceedings of the Sixth (1996) - International Offshore and Polar Engineering Conference*, number 11, pages 251–255. ISOPE, May 1996.

[28] J. Scott Willcox, 1998. personal correspondence.

[29] Dana Yoerger, 1998. personal correspondence.

[30] Dana R. Yoerger, Albert M. Bradley, Barrie B. Walden, Hanumant Singh, and Ralf Bachmayer. Surveying a subsea laba flow using the Autonomous Benthic Explorer (ABE). Technical report, Department of Applied Ocean Physics and Engineering, Woods Hole Oceanographic Institution, 1996.

[31] Zoran Zvonar, David Brady, and Josko A. Catipovic. Adaptive Detection for Shallow-Water Acoustic Telemetry with Cochannel Interference. *IEEE Journal of Oceanic Engineering*, 21(4):528–536, 1996.